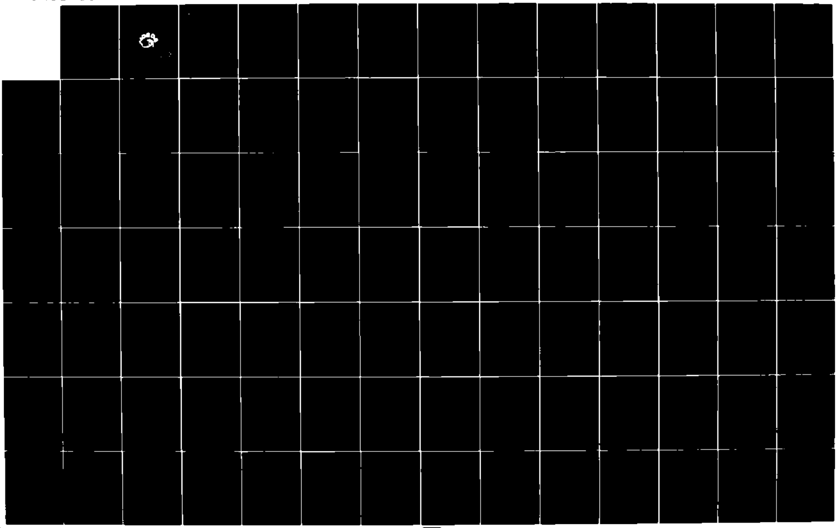
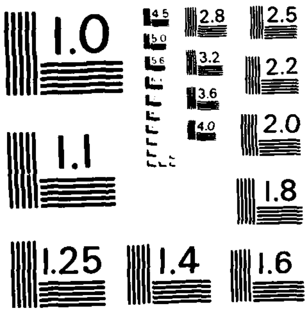
AD-A130 779 A DESCRIPTION OF THE NORDA SATELLITE REMOTE SENSING HARDWARE/SOFTWARE FOR..(U) NAVAL OCEAN RESEARCH AND DEVELOPMENT ACTIVITY NSTL STATION MS.. G D STEPHENSON 1/2

UNCLASSIFIED APR 83 NORDA-TN-205 F/G 9/2 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

NORDA Technical Note 205

Naval Ocean Research and Development Activity
NSTL Station, Mississippi 39529

A Description of the NORDA Satellite Remote Sensing Hardware/ Software for Prospective Users

NAVAL OCEAN RESEARCH AND DEVELOPMENT ACTIVITY
NORDA
1976

Gerald D. Stephenson

Ocean Science and Technology Laboratory
Oceanography Division

April 1983

DEPARTMENT OF THE NAVY
NAVAL OCEAN RESEARCH AND DEVELOPMENT ACTIVITY
NSTL STATION MISSISSIPPI 39529

IN REPLY REFER TO:
335:AEP:bm
3900
Ser 330/242
18 Mar 83

DISTRIBUTION

This information is an introduction to users of the NORDA Interactive Digital Satellite Image Processing System (IDSIPS). It also provides developers of Naval systems which incorporate environmental satellite data with an overview of available software and that which is under development at NORDA. Included are descriptions of the image display hardware, calibration, location, ocean applications and system utility programs, and a few words on the new NORDA satellite receiver capability.

For additional information about this system, please contact the author at Autovon 485-5264, FTS 494-5264 or Commercial 601-688-5264. Potential users of this system or those interested in obtaining copies of software for Naval systems please contact the undersigned at Autovon 485-4864, FTS 494-4864 or Commercial 601-688-4864.

A. Pressman

A. E. PRESSMAN
Head, Remote Sensing Branch
Oceanography Division

DISTRIBUTION:
OPNAV (OP-952)
COMNAVAIRSYSCOM (AIR 370G)
NAVENVPREDRSCHFAC
FLENUMOCEANCEN
NESC (PME 106-48)
COMNAVOCEANCOM (Werner)
NAVOCEANO (Code 9000)

ABSTRACT

This document is intended to provide a general overview of the hardware and software used in the NORDA Code 335 Remote Sensing Branch. Detailed documentation of the International Imaging Systems (I^2S) software can be found in their System 101 _Image Processing System User's Manual._

A license agreement exists between NORDA and I^2S that prohibits NORDA from making this software available to any other facilities. Any facility that is interested in using the I^2S software would have to make a license agreement with I^2S.

ACKNOWLEDGMENTS

Thanks are extended to A.E. Pressman, R.J. Holyer, and E.B. Arthur for their suggestions and assistance in preparing this document. This work was funded under program element 63704N, A.E. Pressman, Program Manager.

CONTENTS

A DESCRIPTION OF THE NORDA SATELLITE REMOTE SENSING HARDWARE/SOFTWARE FOR PROSPECTIVE USERS

1.0 INTRODUCTION

The purpose of this document is to provide an overview of satellite image processing hardware/software/techniques that are operational or in the development phase. A few of these may be applicable for use in other Navy image processing systems.

1.1 COMPUTER SYSTEMS

1.1.1 INTERACTIVE DIGITAL SATELLITE IMAGE PROCESSING SYSTEM (IDSIPS)

The IDSIPS system was installed in late 1977. This system has since been upgraded to consist of the following:

- HP 3000 Series III CPU 1 M Byte Memory
- 1 - 50 M Byte HP Disc Drive
- 4 - 300 M Byte AMPEX Disc Drives
- 1 - HP 1600 BPI 45 IPS Tape Drive
- 1 - Qualex 800/1600/6250 BPI 125 IPS Tape Drive
- 1 - Dunn Instruments 631 Color Camera System
- 1 - Tektronix Dry Silver Paper Recorder
- 1 - EMR Laserfax (GOESTAP)
- 1 - International Imaging Systems (I^2S) Model 70 E Image Analysis Systems
- 2 - International Imaging Systems (I^2S) Model 70 F Image Analysis Systems
- 1 - 300 LPM Printer
- 1 - Operators Terminal
- 8 - CRT Terminals (3 used with I^2S displays)

1.1.2 SATELLITE RECEIVING, PROCESSING, ANALYSIS SYSTEM (UPGRADE)

A turnkey system to receive/process TIROS, GOES, and DMSP satellite data will be installed and operational in November 1983. Data analysis functions will be performed on the system on a noninterference basis with satellite reception/processing. The initial image analysis system will consist of the following hardware:

- Gould SEL 32/27 CPU 1 M Byte MOS Memory
- 1 - SEL Floating Point Accelerator
- 1 - SEL Logic Memory Support Package
- 3 - 300 M BYTE CDC Disc Drives
- 1 - STC 800/1600/6250 125 IPS Tape Drive
- 1 - DECWRITER III Printer
- 1 - HP 8 Pen Color Plotter
- 1 - Muirhead K560 Hardcopy Device
- 1 - 300 LPM Printer

- 2 - CRT Terminals
- 2 - International Imaging Systems (I^2S) Model 75 Image Display
- 1 - 8-Line Asynchronous MUX

The system will be delivered with software to perform the following functions:

- Unattended operation
- Antenna Positioning and control
- Formatter setup
- Data acquisition
- Antenna pointing throughout pass
- Data extraction and geographical correction
- Real-time product generation (hardcopy and Model 75 monitor)
- Data management and archiving
- Registration and gridding
- Hardcopy plotter products
- Orbit prediction
- Radiometric calibration
- Device control
- System resource accounting
- Interactive Image Manipulation and Display (I^2S Sys 575)
- World data base
- Diagnostics and testing

1.1.3 GEOSAT-A

The system to support the GEOSAT-A Ocean Application Program (GOAP) project and process GEOSAT data will become operational in mid-July 1983. This initial system will consist of the following:

- Gould SEL 32/2750 CPU 1 M Byte MOS Memory
- 1 - 300 M Byte CDC Disc Drive
- 1 - STC 800/1600/6250 125 IPS Tape Drive
- 1 - 300 LPM Printer
- 2 - CRT Terminals (1 for operators console, 1 for analyst work station
- 1 - 8-Line Asynchronous MUX
- 1 - International Imaging Systems (I^2S) Model 75 Image Analysis System

1.2 IMAGE ANALYSIS SYSTEMS

1.2.1 I^2S MODELS 70/75

The Model 70/75 Image Computer and Display Terminal is a versatile interactive digital image processing system. In addition to color display of processed multi-spectral or multi-temporal imagery, the Model 70/75 provides three specialized

high-speed Pipeline Processing Channels which significantly off-load the host computer.

The use of local processing in the Model 70/75 does not compromise in any way the repeatability of operations and setups. All control operations, function memory loadings, etc., are nandled through the host computer.

Some of the features and capabilities of the Model 70 and 75 are:

- Pipeline Display Processor to handle arithmetic functions in parallel for each primary color (red, green, blue).
- The Model 70 can be configured with up to twelve channels of image Refresh Memory, any combination of which can be assigned to any primary color (i.e., the sum of two channels can be the final red component, etc.). The Model 75 can be configured with up to 16 channels on only 8 boards using 64K RAMS.
- RS-170 Video input channel for real-time video digitizing.
- Separate Input Function Memory provides scaling, negation, equalization, etc., of input data.
- Videometer hardware to compute histograms (in 67 ms) on raw or processed images.
- Hardware Adder Array allows real-time arithmetic operations (add, subtract, multiply, divide, etc.) between images.
- Feedback Arithmetic/Logic Unit (ALU) to a 16-bit "Accumulator" channel implements recursive procedures, convolutions, and fast (less than 30 sec) n-band image classification.
- Graphics overlays can be displayed in up to 32K colors.
- Cursors can be dynamically displayed in up to 32K colors.
- Real-time scroll (nondestructive Roam) through all image channels, with or without Zoom.
- Ten-bit Digital to Analog Converters.
- Multiple independent CRT monitors.

- Interprocessor Buffer to permit operation from two host computers, one primary and one secondary.

- Address transform mapper eases programming, reduces the main memory requirements, and speeds processing (Model 75).

These, plus such conventional features as normally found in a display terminal, are:

- Random Access (RAM) Refresh Memory organization allows read or write horizontally or vertically, thus simplifying image transposition.

- Programmable internal microprocessor.

- Up to eight independent planes of Graphics Overlay Memory (Model 70).

- Up to sixteen independent planes of Graphics Overlay Memory (Model 75).

- Internal Character Generator.

- Internal Vector Generator.

- Multiple Programmable-shaped Cursors (64 x 64 array).

- Trackball with function switches for manual inputs.

- Joystick or Tablet control of cursors.

- Hardware Zoom (nondestructive image magnification).

- Three separate Output Function Memories (10 bit-in/out) gives an excess of one million displayable colors.

- Hardware Scroll, independent for each image channel.

These processing features are especially useful in interpretive analysis, since the speeds and response in the Model 70/75 are much greater than in the host computer. Weighted sums, differences, ratios and products may be done on a complete 512 x 512 x n set of images in a matter of a few milliseconds. The process is then repeated 30 times per second at video display rates. This creates a high degree of "interactiveness" which is very important to the interpreter, especially in operations such as ratios where the setting of the clip levels for the ratio determines the success or failure of the operation.

Note that the Model 70 uses three independent Pipeline Processing Channels, associated respectively with the three

outputs that provide red, green, and blue video to the color monitor. Each of these pipelines is fed by all Refresh Memory bands. Therefore, generalized switching and combining processes may be performed, which provide for simultaneously influencing any or all of the three video outputs by any or all of the image bands in the Refresh Memory. (This arrangement is quite superior to that of most image display terminals which provide neither generality in switching nor built-in processing. By associating a refresh memory image band directly with a video output, such terminals impose a much greater burden on the associated host computer and its software-controlled processing.)

Of special significance, an option is available called the Feedback Arithmetic/Logic Unit (ALU), which permits the results of the pipeline processing to be read back into the host computer. Additionally, this feedback loop provides an ALU, which can perform 48 different arithmetic and logic operations between the output of the Pipeline Processing Channels and a 512 x 512 x 16-bit "Accumulator"; these ALU operations take place between the 262 K-element arrays in one frame time (33 msec). Thus, the Feedback-ALU provides a powerful tool in convolution, classification, interpolation, etc., operations.

Another important Model 70 feature is the implementation of an on-board microprocessor. The Programmable Processor option provides a DEC LSI-11 that can directly accomplish tasks such as character and vector generation from data supplied to it by the host computer, and can accomplish interactive tasks such as "zoom" and "roam" via functional program modules down-loaded to it from the host computer. In this way the Programmable Processor acts as a subsidiary (or slave) processor which can accept high-level directives from the host computer, perform the detailed processing within its own independent CPU and memory, and inform the host computer upon completion of the task. Concurrently, the host computer can service other real-time or interactive tasks in a multi-user environment. This feature is not available in the NORDA Systems.

An important feature of the Model 75 is the address transform mapper. This Organization Mapper is used to define and control all transfers to and from refresh memory. Use of this subunit allows the user to rapidly transfer image data stored in formats not directly compatible with the internal layout of refresh memory and increases the transfer rate for many commonly used graphics applications.

Another important feature of the Model 75 is the optional built-in LSI-11/23 microcomputer for stand-alone applications. An 8-slot Q-bus backplane embedded within the Model 75 chassis enables a fully configured turnkey work station to be configured within a single chassis. Configurations are available that support a wide range of discs, mag tapes, floppy discs, streamer tapes, and interactive terminals integrated into a console/work-station environment.

1.2.2 FUNCTIONAL DESCRIPTION

The major functions of the I^2S Model 70F are described in the following paragraphs. The description for the Model 75 is assumed to be the same.

1.2.2.1 Input Function Memory

The Input Function Memory (IFM) is included in the Model 70 to enhance the overall speed of operation of the image processing system. The IFM is simply a programmable look-up table that is applied to the data on its way to the Refresh (or Graphics) Memory. The IFM may also be bypassed so as to have no effect. By using the IFM, however, image data of magnitude up to 13 bits is transformed to numbers of 8 bits or less. The IFM thus allows (at rates of up to two megapixels from the host computer) transformation of the image data to include scaling, negation, logarithms, exponentiation or even histogram equalization. The IFM's scale factor (map) is loaded by the host computer. The purpose of the IFM is to off-load a significant burden from the host computer and to speed up the overall system processing capability by not requiring the data to be scaled or mapped in the host computer. The IFM can also be used in conjunction with the Feedback-ALU (see below) to map feedback data before it is returned to Refresh (or Graphics) Memory.

1.2.2.2 Refresh Memory

Each Refresh Memory channel is a solid-state memory consisting of dynamic RAMs. The Refresh Memory Channels are 512 x 512 by up to 8 bits. Since the Refresh Memories are truly random accessible, this allows the host computer to access any pixel (or bit within a pixel) randomly without the latency associated with discs, shift registers, or CCD memories used in other display systems.

During transfers to and from the host computer addressing may be incremented in either rows or columns (or both); this allows images to be written/read either horizontally or vertically. Another important addressing feature of Refresh Memory is the 32-bit Mux Mode. This feature provides for automatic multiplexing of 32-bit data into or out of four channels of Refresh Memory. Thus, during two-dimensional FFT operations as an example, four channels can be used as interim storage for the 32-bit floating point result. These features provide powerful tools for high-speed transposing of image data, or for using Refresh Memory as RANDOM ACCESS bulk storage during host computer image processing.

Each image can be independently scrolled. Independent scrolling allows gross registration (i.e., excluding rubber-sheeting) of images contained in separate channels. Scroll is also used in conjunction with the Feedback-ALU to accomplish

convolutions and edge-detection operations. (Similarly, the Graphics overlay "channel" can be scrolled independent of Refresh Memories.)

The Refresh Memory is of dual-port design, one port dedicated to the video Pipeline Processing Channels for display, the other dedicated to the "system" for read/write operations with the host computer. This approach allows the fastest possible access to the memory by the host computer, and maintains the video output without disturbance. Pixels may be written into the Refresh Memory at a rate of up to 1.25 million per second, so that the Refresh Memory transfer rate is not usually the limiting element in system through-put considerations.

The Refresh Memory in the Model 70 is designed for noise immunity and memory element interchangeability. These features are achieved by use of a multilayer printed circuit board including a ground plane, and incorporation of large-scale integrated-circuit memory elements (dynamic Random Access Memory (RAM) of the type produced by several manufacturers including Advanced Memory Systems, Intel, Monolithic Memories, National, Nippon Electric Corporation and Texas Instruments). The wide usage and great popularity of this MOS-technology element insures availability of high quality parts for years to come. Further, the design of the memory boards has been proven using memories from each of the manufacturers, so that parts may be interchanged without affecting overall performance.

1.2.2.3 Pipeline Processing Channels

Three parallel Pipeline Processing Channels are provided in the Model 70 to perform array arithmetic for each of the three primary colors (red, green, and blue). Each of these Pipelines is fed by the 10 MHz data streams from each of the Refresh Memory channels (and from the Video Digitizer output, which can be considered a "pseudo refresh memory channel"). In this way, generalized switching and combining processes are possible; that is, any Refresh Memory channel (or any combination of Refresh Memory channels) can be assigned to any of the pipelines (which, in turn, supply the RGB primary color outputs of the Model 70). Each Pipeline contains independent X and Y Scroll for each Refresh Memory channel, hardware Zoom, hardware Split Screen, independent Look Up Tables for each Refresh Memory channel, an Adder Array, Minimum-Maximum, Range and Constant array, and an Output Function Memory.

Each of the Pipeline Processing Channels provides for real-time arithmetic operations between the 512 x 512 x 8-bit image arrays contained in the Refresh Memory channels. The Pipelines allow addition, subtraction, multiplication, and division to be performed between the image arrays at a 10 MHz rate, i.e., a new arithmetic operation is performed every 100 nanoseconds as the data flows through the Pipeline.

To understand the basic arithmetic capabilities of the Pipelines, consider only the Look Up Tables, Adder Array, and Output Function Memory. These cascaded *function memories* with the adder between them, allow the four basic arithmetic processes to be implemented at the 10 MHz rate. Once the pipeline has been "programmed" by the host computer, the arithmetic operation is repeated 30 times per second between the 512 x 512 x 8-bit image arrays stored in Refresh Memory.

Addition of the arrays of numbers stored in Refresh Memory is accomplished by loading unity transforms (1:1 scale factors) in the Look Up Tables and the Output Function Memory; the Adder Array then takes the sum on a pixel-by-pixel basis as the image arrays flow through the Pipeline. The resultant array sum is then presented on the RGB Monitor via the Digital-to-Analog (D-to-A) Converter, or may be deposited in a "buffer memory array" (Refresh Memory) for subsequent return to the Host Computer, or reintroduced into the Pipeline for further iterative processing.

Since the Adder Array accomplishes a two's complement sum, subtraction is programmed by loading a positive unity transform in one Look Up Table, a negative unity transform (i.e., sign bit set) in the other Look Up Table, and a positive unity transform in the Output Function Memory.

Array multiplication and division is accomplished via logs and antilogs in the tables. Multiply consists of loading positive log transforms in the two Look Up Tables and loading an antilog (exponential) transform in the Output Function Memory. Division consists of loading a positive log in one Look Up Table, a negative log in the other Look Up Table, and the antilog in the Output Function Memory.

1.2.2.4 Look-Up Tables

Three 256 x 9-bit Look Up Table memories (LUTs) are provided with each Refresh Memory Channel. If the Video Digitizer option is included, three LUTs are also provided with this "pseudo refresh memory channel." Therefore, there is a Look Up Table for <u>each channel</u> of Refresh Memory <u>in each of the Pipelines</u>. These LUTs are one of two programmable processing elements following the Refresh Memories. By providing two levels of function memory following the refresh memory, the four basic arithmetic processes may be implemented at the 10 MHz video data rate.

The data (scale factor) for the LUTs is loaded via the Host Computer. Nine-bit numbers are stored in the Look Up Tables so that two's complement arithmetic may be performed at the output, thus implementing subtraction and division. Also, the output data stream from each LUT can be selectively enabled/disabled; since there are three LUTs for each Refresh Memory channel (one in each Pipeline), this provides for assigning any channel of

Refresh Memory (or any combination of channels) to any of the RGB Pipelines. Hence, complete combining flexibility is provided in assigning image bands to primary colors.

Because it allows for independent radiometric correction for each channel, the Look Up Table also provides the important function of displaying any band in any color, thus allowing the display to approach the ideal of displaying a scene as it originally appeared, not as it is represented on film by a camera. The LUTs also provide the capability for a three-class Look Up Table Classifier, which can be used for preliminary studies.

1.2.2.5 Adder Array

This array digitally takes the two's complement sum of the Look Up Table outputs. Three sets of Adder Arrays are included, one for the red, one for the green and one for the blue Pipelines. The numbers resulting from this addition process may reach 13 bits, if twelve 8-bit Refresh Memory channels are added together.

1.2.2.6 Output Function Memory

Each Pipeline contains an Output Function Memory which transforms the outputs of the Range Registers (see below) to generate the final red, green and blue data streams. Ten bits, or 1024 levels, was chosen as a practical limit for the number of input levels to each OFM. If a Model 70 is maximally configured with 12 channels, each of 8 bits, the numbers passed to the OFM could be as large as 13 bits, or 8192 levels. This range of numbers is accommodated by the Range Register function of the Pipeline, providing for shifting of the video data stream by up to 3 bits.

Each Output Function Memory consists of a 10-bit-in, 10-bit-out high-speed RAM; data (scale factor) for the OFMs is loaded by the host computer. Ten bits of resolution at the output of each Pipeline provides for several eventualities, some of the more important being:

(a) Ten-bit precision during <u>each</u> interaction when accomplishing iterative operations using the Feedback-ALU;

(b) Compensation for gray-scale nonlinearities throughout the total image processing process, from recording sensor to the display monitor;

(c) Compensation for nonlinearities in the monitor reproduction; and

(d) Extended capability that will not be made obsolete by the development of new monitors with wider dynamic range than those now available.

1.2.2.7 Min-Max Registers

The Min-Max Registers examine the 13-bit data stream as it emerges from the Adder Array, and determine the dynamic range of the data by finding the minimum and maximum pixel values. The Min-Max values are read by the Host Computer, and are used in determining how to set the Range Register to process the desired 10 bits via the Output Function Memory.

1.2.2.8 Range and Constant Registers

The Range Registers are used to reduce the 13-bit data stream from the Adder Array to a 10-bit stream for application to the Output Function Memory. The Range Registers allow for selection of ten contiguous bits from the 13-bit data stream (i.e., bits 0 thru 9, 1 thru 10, 2 thru 11, or 3 thru 12). The Constant Registers provide for the addition or subtraction of a 13-bit constant from the data stream before it is applied to the Range Registers. This constant addition/subtraction and right shift selects the desired 10 bit dynamic range of data, and provides for biasing the data to be processed through the Output Function Memories.

1.2.2.9 Hardware Zoom

The Hardware Zoom feature allows magnification (via pixel replication) of the displayed image by a factor of 2, 4, or 8 around an arbitrary location. The specification of the center point of the area to be magnified and the magnification factor is accomplished via the Host Computer. Note that Zoom is nondestructive in that the original data in Refresh Memory is not modified. The Hardware Zoom feature allows any rectangular image subsection in a Refresh Memory channel to be magnified and processed/displayed as a 512 x 512 image. Interpolative Zoom (bi-linear) can be accomplished in conjunction with the Feedback-ALU.

1.2.2.10 Hardware Split Screen

The Split Screen feature enables the display of half of one image band (Refresh Memory channel) in the top or left half of the monitor, and half of a different image in the bottom or right half of the monitor, respectively. This feature permits, for example, comparison of the top halves of two different images.

Split Screen provides the capability of splitting the display into four (equal or unequal) quadrants around an arbitrary location specified by the Host Computer. By using Scroll, each quadrant can then be supplied by any area of any Refresh Memory channel. Thus, Split Screen allows the selection of any subsection of an image resident in multiple memory channels; if twelve channels were implemented, then Split Screen would allow "roaming" around the 2048 x 1536 pixel image database with a

512 x 512 window, and extracting any subsection for processing/displaying. If Zoom is envoked, then a 256 x 256, 128 x 128 or 64 x 64 subsection can be extracted and magnified to a 512 x 512 image for further processing in the Pipelines.

1.2.2.11 Color Monitor

A Conrac 5211 or 5411 Series RGB Monitor is used as the standard display output because of the following features:

(1) Full color or monochrome presentation,

(2) 19-inch diagonal screen size resulting in a displayed image of approximately 11 x 11 inches,

(3) Flicker-free high quality presentation (at 60 fields/second),

(4) Spot size and video circuitry consistent with the high quality required by TV broadcast studio equipment.

The Conrac Series 5211 is a high quality RGB format color monitor. The 5211 has a 19-inch diagonal screen size, operates at 551 (or 525 or 625) TV lines resolution, and offers the advantages of performance unexcelled by any other conventional color monitor. The monitor allows for display of the full 512 lines of data, and effectively eliminates Moire pattern disturbance. Flicker-free performance at nominal brightness and contrast levels is guaranteed by operating the monitor at 30 frames/second (60 fields/second), witn a 2:1 interlace.

Optionally, the Model 70 can be configured with the Conrac 5411 high resolution RGB monitor. The monitor features a high-resolution shadow mask CRT and horizontal dynamic focus to completely eliminate Moire disturbance and present a very sharp image display.

1.2.2.12 Graphics Overlays

Up to eight 512 x 512 one-bit Graphics Overlay channels (planes) can be provided in the Model 70. The Graphics planes are treated as an additional Refresh Memory channel for the purposes of reading and writing via the Host Computer, thereby allowing any bit (or pixel) to be accessed randomly. The eight Graphics planes form an 8-bit data stream, which is supplied, along with the 1-bit Cursor data stream, to the Graphics Data Multiplexer. Under program control this Mux can select the seven least-significant Graphics planes and the Cursor "pseudo plane", or all eight Graphics planes, for application to the Color Assignment Function Memory.

The Graphics Mux also provides for programmatic selection of one of the eight Graphics planes to drive the Status D-

to-A Converter. Thus, one Graphics plane can drive a Status Monitor to display alpha-numeric data, histogram plots, etc. The Graphics Mux also provides for programmatic selection of one of the Graphics planes to provide the Region of Interest (ROI) input to the Videometer and the Feedback-ALU. This ROI is a spatial pixel map used to restrict processing to a "region of interest" by these two processing elements.

1.2.2.13 Programmable Cursor

The Cursor provided with the Model 70 can be considered a "pseudo Graphics plane"; the one-bit Cursor data stream is treated just like the other Graphics planes by the Graphics Color Assignment Function Memory. All pixel values in this pseudo plane are considered to be zero (i.e., off) except within the Cursor RAM.

The Cursor RAM is a 64 x 64 x 1-bit memory. The desired Cursor shape is stored in this RAM by the Host Computer, thereby giving complete flexibility in defining the actual Cursor shape within this 64 x 64 space. Position of the Cursor RAM within the 512 x 512 space of the "pseudo Graphics plane" can be controlled by the Host Computer, by the Trackball, or by the Tablet.

The Host Computer can command the Cursor position or read back the Cursor position at any time. The Cursor can be displayed constantly, blinked at three preselected rates, or turned off. The Host Computer can also selectively link the Cursor X and Y position registers to the Trackball unit.

1.2.2.14 Trackball and Tablet

The Trackball unit is used to selectively control the Cursor position in the X or Y direction (independently or simultaneously). The Cursor can be linked/unlinked from the Trackball in X or Y by the Host Computer. When the Cursor is moved via the Trackball (or Tablet), a status bit is asserted to the Host Computer; most Host Computer interfaces allow this bit to generate an interrupt. The Trackball has been designed to allow the user to move the Cursor in one pixel increments.

Four function buttons are provided on the Trackball housing. When pushed, the buttons cause a status bit to be asserted to the Host Computer; most Host Computer Interfaces allow this bit to generate an interrupt. The button activation is stored in a register, and an audible "beep" is generated when this register is read by the Host Computer. This register and the Button Pushed status bit allows the Host Computer to use the buttons in either an interrupt or noninterrupt (polling) mode. When used in an interrupt mode, the four buttons provide four distinct states; when used in the polling mode, sixteen possible states can be obtained.

A Summagraphics ID-11 Tablet with Stylus (pen) can also be used to control the Cursor position. This 28 x 28 cm (11 x 11 inch) active surface Tablet is linked to the Cursor position registers by the Proximity status bit from the Tablet. Proximity is asserted when the Tablet's pen is within 4 mm (5/32 inch) of the active surface. A Z-Axis status bit from the Tablet is asserted when the pen is pressed onto the active surface (note that a sheet of paper, etc., can be between the pen and the active surface). Proximity and Z-Axis status bits are available to the Host Computer. Therefore, with Proximity only asserted, the Cursor could be positioned to the beginning of a line, and when Z-Axis was asserted the line could be copied into a Graphics plane. In this way the Tablet could be used to trace map overlays, etc.

Approximately 20.5 x 20.5 cm (8 x 8 inch) of the Tablet's active surface is required to control the cursor position in the 512 x 512 space of the displayed image; the remaining area of the Tablet surface could be used for menu overlays.

1.2.2.15 **Color Assignment Function Memory**

The 8-bit data stream from the Graphics Mux is applied to the Color Assignment Function Memory. The purpose of this function memory is to assign one of the possible 32, 768 colors to each Graphics plane, and to dynamically change the assigned colors as the Graphics planes overlay (intersect) each other. In this way, all Graphics Overlays (and the Cursor) can always be seen.

The Color Assignment Function Memory has 256 locations of 16-bits each. The least-significant 5 bits of each location define the intensity value to be used for the blue component of Graphics color; the next 5 bits define the green component, and the next 5 bits define the red component. The most-significant bit of each location defines an overlay/insert (replace/add) mode. That is, the Graphics data can replace the image data, or it can be added to it. Note that the 5 bits of Graphics data are added to the 5 most-significant bits of the image data; if an overflow condition occurs, then the resultant 10 bits to the D-to-A converter are latched to a maximum (saturated) value.

The Host Computer can thus program the Graphics colors by loading a map into the Color Assignment Function Memory. This map can thus define what color is to be displayed when any one Graphics plane is on, and also define a different color to be displayed for each of the other 248 possible combinations of on/off conditions of the eight Graphics planes. Thus the spatial areas where the Graphics planes overlap each other can be coded to a different color from the 32K colors available. The ability to dynamically change color assignments for overlapping regions guarantees that all Graphics Overlays (and the Cursor) can be distinguished from other Graphics overlays at all times.

1.2.2.16 Videometer

The Videometer is a processing element that computes the histogram of the processed data streams just prior to their conversion to video (at the output of a Pipeline). The implementation of histogram generation in hardware provides three significant advantages over software implementations.

First, the Videometer hardware has the ability to histogram the results of the Model 70's processing (after the data has been processed with the Pipeline Processing Channels). (Note that this histogram cannot be precomputed from the histograms of the original data inputs in the general case.) Second, the Videometer can obtain the histogram of the entire image, or of any subarea of the image as defined by an image mask (Region of Interest) loaded in a Graphics plane. Third, the Videometer computes the histogram rapidly (in 67 msec on the full 10-bit data out of the Pipelines) without imposing any computational load on the Host Computer.

Since the histogram is a tool widely used in the field of image processing, several important applications of the Videometer are found in:

(1) Histogram equalization;

(2) Dynamic histogram clipping (percentage clipping);

(3) Automatic scaling for arithmetic operations such as addition, subtraction, multiplication and division;

(4) Automatic scaling for spectral transform operations such as FFT, SLANT, KL and Hadamard;

(5) Training area definition for table-lookup classification; and

(6) Area calculation, i.e., the percentage of the image that lies within specified grey level ranges.

1.2.2.17 Feedback-Arithmetic/Logic Unit

Except for image data scaling performed by the IFM, the various transformations performed in the hardware described above do not actually modify the image data which is stored in Refresh Memory, but merely alter the way in which it is displayed. If it is desired to retain the actual processed 10-bit image data out of the Pipeline, it may be "fed back" to a selected Refresh Memory (via the IFM if desired) in a single frame time (1/30th of a second). Note that the IFM can be extended to 13 bits (on input) when the feedback loop is included in the Model 70.

In addition to providing for the retention of the processed data, this feedback loop also enables the Model 70 to

perform recursive procedures in which the output of the Nth processing step becomes the input to the (N+1)st step. During the feedback process, the entire image can be shifted horizontally and/or vertically any number of pixels. The pixel shifting capability at video rates allows the user to perform a spatial convolution using successive iterations to form the sum of prod-products required in the convolution process.

The Feedback-Arithmetic/Logic Unit can be used to perform arithmetic or logical operations between a Pipeline Processing Channel output (OFM output) and the current contents of the "Accumulator" (Refresh Memory channels 0 and 1); the results of the operation are then stored back into the 16-bit Accumulator, or into any Refresh or Graphics Memory channel. The IFM may be envoked during this feedback operation to scale the ALU output before it is stored in memory.

The ALU features two op-code registers so that two different arithmetic or logical operations can take place during the same feedback cycle. The op-code register selection is controlled by the ROI graphics plane (region of interest) so that one function is accomplished within the region of interest, and a different function is accomplished outside the region of interest.

Prior to the output of the ALU processing logic there is a programmatic selection of what data shall be output. In addition to the data derived from the function programmed from Figure 1-4 (i.e., the ALU data), eleven other data streams can be output by the ALU processing logic. These include one of eight Constants (discussed below), the unmodified current contents of the Accumulator, the unmodified output of the selected Pipeline, or the unmodified data from the External Port input to the ALU. These twelve data streams are available from a "pseudo array." The array is addressed by the Output Select Table (OST). The array is as follows:

ARRAY ADDRESS	DATA STREAM
00	Constant 0
01	Constant 1
02	Constant 2
03	Constant 3
04	Constant 4
05	Constant 5
06	Constant 6
07	Constant 7
10	Accumulator data
11	Selected OFM data
12	ALU data
13	External Port data

The OST is a 3-bit-in, 4-bit-out look up table. Thus, one of eight data streams from the possible twelve streams may be output at any given pixel time; the selection of which stream is to be output depends on the function map loaded in the OST.

The function map loaded in the OST can consist of numbers in the range of 0 through 13 (octal). These 4-bit numbers then address the array described above. The OST look up table's function map is loaded by the host computer.

The address-in port of the OST look up table is derived from the ALU's "equal" output, from the ALU's "carry" output, and from the Graphics "ROI" plane (Region Of Interest). Thus, all combinations of Carry, Equal and ROI can select any one of eight (from the possible twelve) array data streams at any given pixel time within the frame.

For example, if all locations of the OST look up table were loaded with 12 (octal), then all combinations of Carry, Equal and ROI would cause ALU data to be output from the ALU subunit. If, however, locations 0 thru 3 were loaded with 10 (octal) and locations 4 through 7 were loaded with 11 (octal), then the unmodified current output of the selected pipeline would be output by the ALU pipeline within the ROI area (regardless of Equal and Carry states) and the unmodified current contents of the Accumulator would be output by the ALU processing logic outside of the ROI area.

The Constants are loaded or read by the host computer in 16-bit one's complement form.

1.2.2.18 Programmable Processor

(Note: This processor is not available in the NORDA system.)

The Programmable Processor provides a DEC LSI-11/03 microprocessor for further offloading of image processing tasks from the Host Computer. The Programmable Processor can directly accomplish tasks such as Character and Vector Generation from data supplied to it by the Host Comuter, and can accomplish interactive tasks such as Zoom and Roam via functional modules (Primitives) down-loaded to it from the Host Computer. In this way the Programmable Processor acts as a subsidiary (or slave) processor which can accept high-level directives from the Host Computer, perform the detailed processing within its own independent CPU and memory, and inform the Host Computer upon completion of the task. Concurrently, the Host Computer can service other real-time or interactive tasks in a multi-user environnment.

The hardware provided with the Programmable Processor includes:

- Model 7008-06, Q-bus Interface to the DEC PDP-11/03 and cables to the Model 70.

- Model 7017, Inter-Processor Buffer to provide a second computer port to the Model 70 (second port used by the 7008-06).
- 11/03-EA, "Small Chassis" PDP-11/03 with 28 K bytes RAM memory.
- KEV11, Extended Instruction and Floating Point Instruction Sets.
- MRV11-BA, PROM Memory boards.
- MRV11-BC, 1 K x 8-bit PROM Memory chips (can be expanded to a total of 32 K bytes of Read Only Memory).
- The PDP-11/03 is mounted in the Model 70's Electron-Electronics Cabinet.
- Firmware (in PROMs) provided with Programmable Processor includes the following:
- Character Generation; the Programmable Processor accepts an ASCII cnaracter string and parameters defining string location, horizontal/vertical direction, and Graphics plane to be used. A standard (default) character font is included in PROM, but may be replaced by a font down-loaded into RAM by the Host Computer.
- Vector Generation; the Programmable Processor accepts a string of word pairs defining the X,Y coordinates of vector end-points, plus parameters defining write/erase mode and the Graphics plane to be used, and writes (or erases) a line connecting the points specified. Thus, the Vector Generation performs a Vector-to-Raster conversion.
- Primitives Down-load Facility; the Programmable Processor contains a monitor that supervises the receipt and execution of functional program modules down-loaded from the Host Computer. These modules consist of selected Primitive Subroutines contained in the Model 7023 Primitives Software Package.

1.2.2.19 Video Digitizer

This Model 70 feature converts the output from vidicon TV camera (or other RS-170 compatible video source) to a 512 x 512 image at 8-bit resolution with a sampling aperture of 50 pico seconds (maximum). Note that the image digitization is accomplished in real-time (1/30th of a second).

The digitizer data is applied to a Look Up Table in each of the Pipelines; thus the Video Digitizer can be considered

considered as a "pseudo refresh memory channel." The Feedback-ALU can be used to store the digitized image in an actual Refresh Memory channel. The Feedback-ALU can also be used in conjunction with the 16-bit Accumulator channel for noise integration using up to 256 frames of data.

1.2.2.20 **Inter-Processor Buffer**

The Inter-Processor Buffer (IPB) provides a second interface port to the Model 70, thus allowing two host computers to be interfaced to the same Model 70. The IPB also provides a 16-bit data path between the two host computers (or between a host computer and a microprocessor). One processor is termed the Master, and the other processor is termed the Slave. If an access contention to the Model 70 arises, the IPB gives priority to the Master processor over the Slave processor.

The data path features a 256 word (16 bit per word) "Mailbox" buffer memory. The Mailbox can thus be used to pass "messages" between the two processors.

When a processor initiates a write transfer to the Mailbox, the "Done" status bit to that processor is negated; when that processor completes the write operation of the Mailbox, then the "Attention" status bit to the other processor is asserted. When the other processor initiates a read transfer of the Mailbox, then the "Attention" bit to that processor is removed; upon completion of the read operation of the Mailbox by the other processor, then the "Done" bit is asserted back to the initiating processor. The "Attention" and "Done" status bits are normally used to generate interrupts to the two processors. In this way, software protocol can be established to pass messages between the two processors, or this link can be used to down-load programs from one processor to the other.

1.2.2.21 **Primary Power**

The Model 70 can be configured to operate from the following primary power inputs:

(1) 115VAC, 60Hz, $\pm$10% (standard configuration), or

(2) 230VAC, 50Hz, $\pm$10%, or

(3) 100VAC, 50Hz, $\pm$10%.

1.2.2.22 **Internal Processing Rates**

The standard configuration of the Model 70 provides switch selection of internal processing rates of 525 lines/frame or 551 lines/frame (i.e., 480 displayed lines, or 512 displayed lines); this is at 2:1 interlace, 60 fields/second. Optionally, the Model 70 can be configured to operate at 625 lines/frame, 2:1 interlace, 50 fields/second.

1.2.2.23 NTSC Encoder

The NTSC Encoder converts the video output of the Model 70 to conform to the (U.S.) National Television Standard Code. The encoder converts the separate R, G, B, sync, and blanking video signals to a composite video that is NTSC-compatible. Thus the Model 70 can drive other NTSC-compatible devices such as video tape recorders or video discs. It should be noted, however, that this reduces the vertical resolution to 480 displayed lines, since the NTSC 525 line rate is used (the bottom 32 lines of the stored image are not output by the Model 70).

2.0 DATA SOURCES

2.1 CCT TAPES

CCT tapes containing satellite image data is received from several sourcs. These sources include:

- o LANDSAT--EROS Data Center.
- o TIROS-N/NOAA, CZCS, GOES--NOAA Environmental Data and Information Services, Satellite Services Division.
- o TIROS-N/NOAA--Lannion, France.

2.2 SATELLITE DATA RECEIVING SYSTEM

Data will be acquired at NORDA from the GOES, TIROS-N/NOAA series, and DMSP satellites via the satellite receiving system.

2.2.1 GOES

Near-real-time GOES East and GOES West "stretched" Visible Infrared Spin Scan Radiometer (VISSR) data will be received and processed. The largest area of interest for reception of visible data will be 5000 pixels by 5000 lines at full resolution (0.8 km). Thermal infrared data will be collected at its full resolution (8 km). Telemetry and calibration data will be used for proper geographic registration, radiometric calibration, and quality control.

2.2.2 TIROS-N/NOAA Series

Advanced Very High Resolution Radiometer (AVHRR) data from the TIROS-N/NOAA series satellites will be received in real and near real time. Global coverage of 1.1 km and 4.0 km resolution data will be available on a selected basis. Telemetry and calibration data will be used for proper geographic registration, radiometric calibration, and quality control.

2.2.3 DMSP

Data from the Operational Line Scanner (OLS) aboard the DMSP satellite will be received. Fine (0.6 km) resolution and smooth (2.8 km) resolution in the visible and infrared portion of the spectrum will be available. Telemetry and calibration data will be used for proper geographic registration, radiometric calibration, and quality control.

2.2.4 GEOSAT

Data that is required to support the GOAP Project will be received via 9600 band lines from the APL receiver site. This data will be passed through algorithms to produce products for distribution to other Navy installations.

3.0 CALIBRATION

3.1 CCT TAPE DATA

Satellite image data is calibrated as the data is input from tape and stored in disc files as follows:

3.1.1 EDIS (NOAA6, NOAA7, TIROS-N) AVHRR

The Slope and intercept information from the 5 spectral bands are monitored for any change between successive scan lines. When a change in any selected band occurs, a lookup table for the band is re-computed to calibrate the raw spectral count values. The raw spectral count values for each sample of the selected band/s are then used as indices into the respective table to obtain the calibrated output value.

3.1.2 LANNION(NOAA6, NOAA7, TIROS-N) AVHRR

To calibrate LANNION formatted image data, lookup tables for selected bands are generated and utilized for raw spectral count value calibration as described in 3.1.1. However, the slope and intercept data is obtained in a different manner. From bands 1 and 2, the slope and intercept values used are pre-defined constants for each of the satellites (NOAA6, NOAA7 and TIROS-N). Slope and intercept values for bands 3, 4 and 5 are derived from the PRT, black body video and space video data over 4 scan lines. These averages are then used to compute slope and intercept values.

3.1.3 EDIS (NIMBUS-7) CZCS

This function reads and processes CZCS image tapes. The tapes must be formatted as described in "NIMBUS Observation Processing System (NOPS) Tape Specification T744041 CZCS CRT Tape." This function creates calibrated or uncalibrated images from these tapes. Optional output files are control point files

and documentation files. The control point files are created using the anchor points found on CZCS tapes. The documentation files are the trailer record found at the end of each CZCS image data file. The calibration is done as follows:

For Channels 1-5

```
outval = Ifix(((Count * Slope) + Intercept) * 100)
Count - data value from tape
```

For Channel 6

```
outval = Ifix((look-up * 10) + 500)
Look-up - is the Celsius degree associated with the
          count value found on the tape. A look-up
          table from the tape is used to compute the
          temperature.
```

Some tapes do not contain the look-up table; when this condition occurs Channel 6 will be computed as normal. No calibration will be done to Channel 6.

4.0 EARTH LOCATION

4.1 CIA WORLD DATABASE WDB2

The coastline data has been extracted from this database and placed in disc files on the IDSIPS system for use by principal investigators. The Cartographic Automatic Mapping Program is available on tape but has not been placed on the system. For our purpose, separate programs were developed for use with the I^2S display system as follows:

4.1.1 MAP

Two programs were developed to access and display on the I^2S Model 70 data extracted from the World Database. The first program, >MAP, is used to access the database and create an image file of the area of interest. The second program, MAP, will draw the coastline directly on a graphic overlay of the Model 70.

4.2 EPHEMERIS

Orbital element data is maintained in files for each satellite of interest to investigators. This data is input to the following programs to assist in image registration.

4.2.1 ORBIT

Program to compute satellite position, sensor coverage and antenna pointing information. An output file may be generated

which contains satellite subtrack or specified sensor/channel coverage. This file may be input to the world coastline data base display program, ^MAP, to display the satellite track and sensor coverage on the I^2S Model 70. The original program was obtained from Otis Brown of the University of Miami.

4.2.2 ORBIT'CNPT

This program was derived from some basic subroutines of the ORBIT program. The program is used to generate control point files for use in image registration. Control points can be obtained for NOAA6, NOAA7, TIROS-N and NIMBUS-7 (CZCS). The EDIS CCT tapes and respective ephemeris file are inputs to this program.

4.2.3 ORBIT PREDICTION

Program will be used to position antenna for specific satellite data ingest. This is in development and will be delivered with the satellite receiving system in November 1983.

5.0 APPLICATIONS

The following paragraphs provide brief descriptions of programs/techniques developed to support requirements of the Remote Sensing Laboratory. Disc files are defined to be in formats that are compatible to all program modules of the I^2S S101 software package.

5.1 PROGRAMS/TECHNIQUES

5.1.1 AVHRR

CCT tapes obtained from EDIS containing NOAA6, NOAA7 and TIROS-N satellite image data are read by this program and stored in image files on disc. The program provides the user with options to select any combination of 1 to 5 image data channels, select an image subsection and calibrate/not calibrate the data as it is input and stored. It also can save a file of lat/long anchor points for subsequent image registration.

5.1.2 LANNION

CCT tapes obtained from Lannion, France, containing NOAA6 and NOAA7 satellite image data are read by this program and stored in image files on disc. This program also provides the user with options to select any combinations of 1 to 5 image data channels, select image subsection and calibrate/not calibrate the data at it is input and stored.

5.1.3 CZCS

CCT tapes obtained from EDIS containing NIMBUS-7 (CZCS) images are read by this program and stored in image files on disc. Options are also provided to select any combination of 1 to 6 image data channels and print/not print a set of anchor points obtained from the CRTT documentation record. The program also provides an option to calibrate/not calibrate the data as it is input and stored. It also can save a file of lat/long anchor points for subsequent image registration.

5.1.4 MCSST

Technique to produce atmospherically corrected sea surface temperatures from multi-channel sea surface temperatures. Special features of the I^2S Model 70 hardware and supporting software are utilized to produce this result.

5.1.5 CLOUD REMOVAL/TIME SERIES

The cloud removal program composites several registered images (assumed to be GOES IR images of the same region at different times) to produce an output image. The compositing process consists of inserting the lowest count value (corresponds to the highest temperature) from all input images into each of the pixel locations of the output image. Since sea surface temperatures are normally warmer than cloud top temperatures, this procedure effectively removes clouds from all areas that do not remain cloud covered during the entire compositing period.

5.1.6 DIGITAL FILTERING

Program to normalize and smooth data and provide edge enhancement.

5.1.7 LCOPY

This program module was derived from the I^2S S101 module, copy, to provide a means of obtaining accurate image registration. This technique is as follows:

The satellite scanner, such as AVHRR, CZCS, etc., scans at a constant angular velocity and digitizes the signal at equal intervals of time. Thus, each pixel represents a fixed increment in scan angle. A pixel does not, however, represent a fixed interval in ground distance. This is shown graphically in Figure 1.

In Figure 1, a 10° increment in scan angle at nadir (θ_1) subtends an earth surface distance D1, while the same 10° scan angle increment off nadir (θ_2) subtends a much larger earth surface distance, D2; hence the large edge distortion.

We have developed a new approach called scan line linearization, which essentially amounts to a resampling of the scan line. In the new sampling scheme, each pixel represents a fixed increment of earth surface distance rather than a fixed increment in scan angle. A scan line linearized in this way does not contain the severe edge distortion since a pixel represents a fixed distance on the ground whether that pixel is at nadir or at the the edge. Linerization, therefore, transforms the satellite into a low distortion case that falls well within the intended capabilities of the IDSIPS WARP function.

To overcome this distortion, we linearize each scan line in the image and then WARP the linearized result. Of course, since we have resampled the image data, control points files established for the original data are not valid for the linearized image. The control points files must, therefore, be processed through the same linearization algorithm applied to the image before they can be used to warp the linearized image. The new IDSIPS function LCOPY has been written to linearize both image files and control point files.

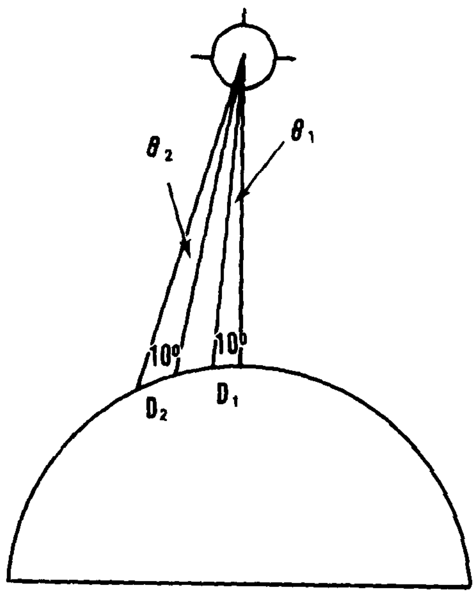


FIGURE 1

5.1.8 TIP

A program will be developed to decommutate the Tiros Information Processor (TIP) data from the AVHRR data stream received by the satellite receiving station.

5.1.9 VLOOP

Program to allow the user to flicker between an array of images each assembled in one or more I^2S model 70 refresh memories. Thus providing a technique for comparing time-sequence series of phenomenon such as movements of ocean mesoscale features. This original I^2S program was modified to provide an option to read/display temperatures from GOES subimages.

5.1.10 DTEM

Designed as a subroutine called by VLOOP and TEMD. The routine reads the pixel value from the image and computes the corresponding temperature for the specific point. This temperature value is printed on the video screen as part of the cursor. The cursor position, unadjusted pixel count value and temperature are printed on the terminal screen.

5.1.11 TEMD

Developed to use the subroutine DTEM to compute and display temperatures from any image. The image may be multi-banded.

5.1.12 GRID

This program was developed to generate a mercator or regular map grid for overlaying TIROS image data on the I^2S Model 70 display. The resultant grid is created by the program from these inputs; latitudes and longitudes corresponding to upper left corner of the grid image, pixel/degree ratio, grid size in pixels, grid tic spacing value, type of projection required.

5.1.13 MCOPY

Program derived from the I^2S "COPY" program. This version enables the user to scale control points in a control point file during file copy operations.

5.1.14 GOESCONTROL

Program designed to generate a file of control points from GOES tape images received from EDIS. The user specifies the grid the image is to be warped to and the area of the image from which control points are to be selected. The latitude and longitude of these points is computed and the values scaled to a mercator or equirectangular grid.

5.1.15 MERCONV

Program to rescale, shift, convert to mercator and combine control point files. The program is used to prepare control point files for input to the I^2S WARP program.

5.1.16 LOWTRAN

Atmospheric transmittance model originated by the Air Force Geophysics Laboratory. This program is used for predicting atmospheric transmittance and thermal radiation emitted by the atmosphere and earth from 350 to 40,000 cm^{-1} at a spectral resolution of 20 cm^{-1}.

5.1.17 TWOSAT

Program to take co-registered GOES and polar orbiter images and generate an atmospherically corrected SST image. Cortion is based on differential absorption resulting from differences in atmospheric path length.

5.1.18 GEOSAT

Program to process altimetry data using SEASAT data to simulate GEOSAT data. The program produces residual sea height vs. location, latitude and longitude, plots tracks of altimeter data along the satellite track on the I^2S model 70 graphics to overlay the data image.

5.1.19 CLEAR WATER RADIANCE

This program computes clear water radiance values from CZCS data. These values are then input to the program to Remove Atmospheric Effects 5.1.20.

5.1.20 REMOVE ATMOSPHERIC EFFECTS

This program outputs an atmospherically corrected CZCS image from calibrated CZCS data and radiance values computed by the Clear Water Radiance Program 5.1.19.

6.0 USING THE IMAGE PROCESSING SYSTEM

Each system user is assigned a group name and user name for his/her use when logging on to the system. A catalog and tape directory for image file maintenance are also assigned to each user. The user information described in the following paragraphs is oriented to the IDSIPS system; however, similar techniques will be used on the GOULD SEL systems.

6.1 IMAGE PROCESSING OPERATING SYSTEM

Image Processing Operating System (IPOS/101) is a special purpose, disc-based software system which schedules and controls image processing programs that run on I^{2}S System 101.

IPOS/101 operates under the control of the Hewlett-Packard MPE operating system and provides an easy to use command language interface. With little or no knowledge of the underlying programming structure, you can use IPOS commands to process any image data. Your interaction with IPOS (Fig. 1-1) causes MPE to use the host computer (HP 3000), the Image Computer (I^{2}S Model 70), and the system peripherals to initiate the image operation.

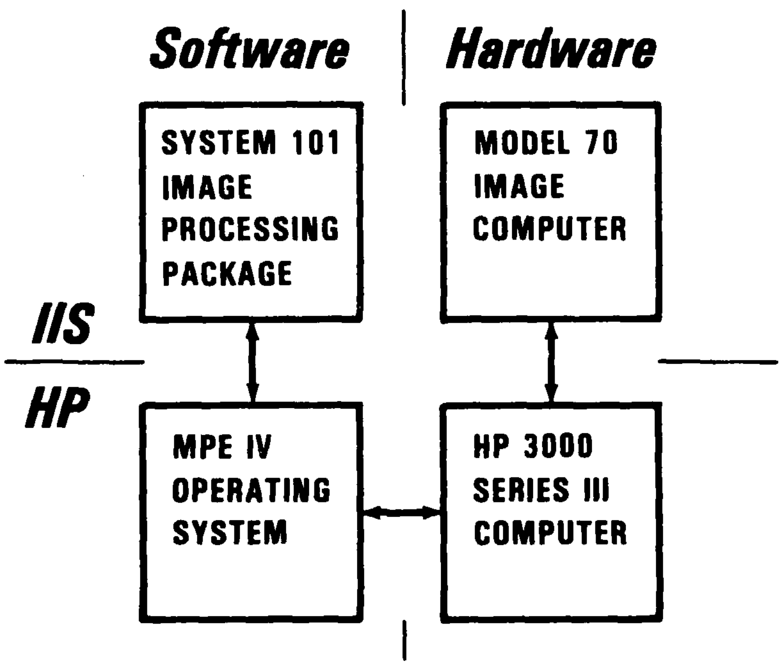


Figure 1-1. System 101 Block Diagram.

So that you can communicate with the System 101 to initiate an image processing interactive session or a batch job you should first log on to an account containing the System 101 software. The manner in which you conduct sessions or jobs is documented in the MPE Commands Reference Manual. Your log-on merely indicates you are in communication with MPE. The actual image operation begins with the following command that loads and executes the program, CI (Command Interpreter), and thereby activates IPOS:

```
:RUN CI.PUB;LIB=P
```

Many installations use the HP MPE IV User Defined Commands so that CI is run automatically when the user logs on.

IPOS allows you to enter commands whose choice is determined by the nature of the particular image application. Each IPOS command is first accepted and parsed by the IPOS Command Interpreter which links it to the appropriate system procedure. IPOS dynamically allocates such system resources as pipeline processing channel(s), feedback-ALU, peripheral devices, refresh memory channel(s), input function memory, and output function memory to each image application program as required. Notice that all discussion related to image data processing applies to one or more users in the multi-user environment of System 101.

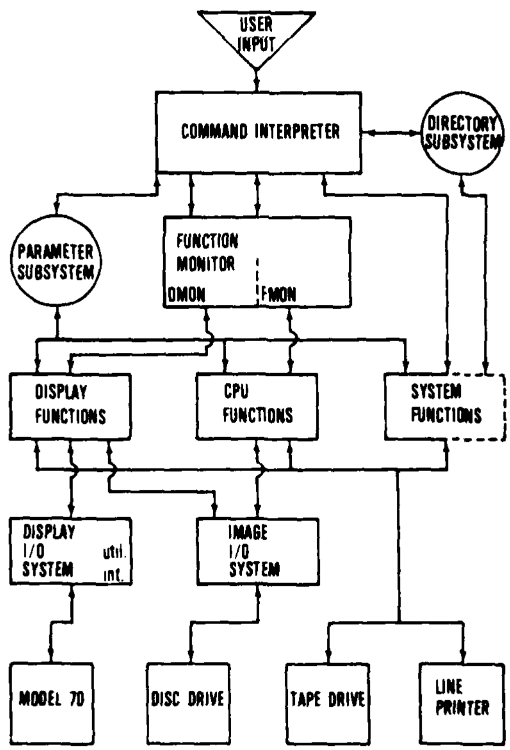


Figure 1-2.

The major components of IPOS, depicted in Fig. 1-2, are described in the following paragraphs:

6.1.1 COMMAND INTERPRETER

The Command Interpreter functions as the main control program for IPOS. It resides in the PUB group of your account. After its loading and execution, the Command Interpreter prompts you for your first command. The prompt consists of a line number, initially one, followed by four blank character positions. The command string is entered after the prompt. You must terminate every IPOS command with a semicolon;otherwise,a carriage return is treated by the command interpreter as a continuation of the same command to the next line. The line number is increased by 0.01 in the case of multiple line commands. The Command Interpreter continues to prompt for additional lines until it encounters a semicolon indicating the end of command. You may enter more than one command in the same line. For example,

^ACQUIRE (UNIT=2); IMAGE^DISPLAY;

After you type in the semicolon followed by carriage return, the Command Interpreter checks the syntax of the command. You get a message on the standard output device if an error is detected. For example, the message,

INVALID DELIMITER MIX

appears if you forget to type the delimiter \ before typing the system command, LISTCAT. You are also provided with error recovery procedures and a repeated prompt for you to re-enter the command. If the command has references to image data sets, the Command Interpreter checks to see that the input image names are in the appropriate catalogs and the output image does not preexist. The Command Interpreter also verifies that the function specified in the command is present in the function library. Errors detected during these tests are also conveyed to you in the form of messages on the standard output device.

After determining the validity of the command, the Command Interpreter builds the internal tables needed by the process(es) that execute the command. The process enters one of the three possible queues depending on the type of function (system, display processor, or CPU processor) specified in the command.

The Command Interpreter executes the command if the command specifies a system function (Section II). The Command Interpreter transfers control to the Function Monitor or the Display Monitor depending if the specified function is a central processor function (Section III) or a display processor function (Section IV). After the initiation of the program, the Command Interpreter prompts you for the next command, with the line

number incremented by one. The next command goes through the same sequence of operations as the previous command. If the queue is empty, the command is executed immediately;otherwise,it waits until all the previous commands are initiated. The multi-user environment of the System 101 provides a set of three queues for each user of the system.

The commands constitute the functional heart of IPOS. To accomplish any image processing task, you need to use one or more IPOS commands. As mentioned earlier, you should enter a command in response to the prompt issued by the Command Interpreter. All the commands have the following format:

[IN1[IN2..[IN10]]..] \^> funcname[paramslist][\^> OUT1[OUT2..[OUT10]]..];

Notice that only the initial delimiter (\ ^ or >), the name of the application function, and the semicolon are required in each command. The remaining elements of the command format are optional and their selection in each case depends on the definition of the particular command. All the IPOS commands are classified on the basis of the specified function type as indicated below:

System Commands (identified by the delimeter \)

Display Processor Commands (identified by the delimiter ^)

CPU Processor Commands (identified by the delimiter >)

You should use the appropriate delimiter for each command; otherwise,your command is rejected. For example, the command,

>ACQUIRE:

is not acceptable to IPOS, as the correct delimiter for the display processor function ACQUIRE is ^. Also note that this command uses neither an input image nor an output image. The A\DELETE; command, which deletes the image A, specifies only the input image. If more than one image is specified, they are separated either by blank(s) or commas. For example, the following commands are equivalent in meaning:

A B > ADD > C;
A,B > ADD > C;
A ,B> ADD > C;

The names of images, parameters, and functions must not contain blanks, although single quotes (') are allowed to make the names more readable. For example, SOUTHERN'REGION is easier to read than SOUTHERNREGION.

You may use abbreviations for command names. For example, FAST and SEL are used instead of the function names, FAST'DISPLAY and SELECT in the following examples:

```
GIRL^FAST;
$A(1)^SEL;
```

You can refer to an image cataloged in some other user's catalog provided you have the appropriate access to the particular catalog. In the following example, the image, REDSQUARE, is cataloged in the image directory of the user, USER2.

```
REDSQUARE.USER2 ^ FAST;
```

6.1.1.1 Specifying Image Subsections

If you wish to specify only a portion of the input image (subimage), use the following subimage descriptor:

```
inimage (SS SL NS NL SI LI:BL:TL)
```

where:

SS = starting sample (column) number of the image array
SL = starting line (row) number of the image array
NS = number of samples
NL = number of lines
SI = subsampling increment in the samples dimension
LI = subsampling increment in the lines dimension
BL = list of the selected bands (for multispectral band)
TL = list of the selected times (for multitemporal images)

For example, consider the input image, SCENES. It is constructed from two LANDSAT pictures of a geographical area taken on two different occasions. SCENES contains 3240 samples, 2340 lines, and 4 bands. The function, COPY, is used to produce the output image of SCENES. The following command produces an exact copy, SCENES'NEW (output image), of the input image, SCENES:

```
SCENES > COPY > SCENES'NEW;
```

The same output image, SCENES'NEW, is produced if the subimage format for SCENES is used by specifying the original values of the modifiers:

```
SCENES (1 1 3240 2340 1 1: 1 2 3 4: 1 2) > COPY > SCENES'NEW;
```

6.1.1.2 Spatial Subsectioning

The following command selects the first 1000 samples and 500 lines from the upper left-hand corner of SCENES to produce the output image, UPPER'LEFT.

```
SCENES ( 1 1 1000 500) > COPY > UPPER'LEFT;
```

The same output image, UPPER'LEFT, is created by specifying the original values of the remaining parameters:

```
SCENES(1 1 1000 500 1 1: 1 2 3 4: 1 2) > COPY > UPPER'LEFT;
```

6.1.1.3 **Spatial Subsampling**

The following command reduces the size of SCENES by selecting every fourth sample and every second line;

```
SCENES (1 1 3240 2340 4 2) > COPY > SMALL;
```

The output image, SMALL contains 810 samples, 1170 lines, 4 spectral bands, and 2 temporal copies.

6.1.1.4 **Spectral Subsectioning**

The following command selects only bands, two and three, of the two LANDSAT scenes, and produces the output image, BANDS. BANDS contains 3240 samples, 2340 lines, the second and the third band, and two temporal copies.

```
SCENES (:2 3) > COPY > BANDS;
```

Band one of BANDS corresponds to band two of SCENES and band two of BANDS corresponds to band three of SCENES. The order of the spectral bands can be changed:

```
SCENES (:4 2) > COPY > DIFFERENT'BANDS;
```

Band one of DIFFERENT'BANDS corresponds to band four of SCENES and band two of DIFFERENT'BANDS corresponds to band two of SCENES.

6.1.1.5 **Temporal Subsectioning**

The following command produces only one temporal copy corresponding to the second temporal copy of SCENES:

```
SCENES (: : 2 ) > COPY > ONLY'ONE;
```

6.1.1.6 **Parameter Subsystem**

The Parameter Subsystem is a group of subroutines that allows you to specify parameters to the various programs. Parameters are enclosed in parentheses following the function name. For example,

```
A B > ADD (WEIGHTS=.5, .5)>C;
```

You should specify parameter values either in the commands or as responses to the prompts issued by the application programs. Prompts are also issued if you make mistakes while entering parameter values or if you omit a mandatory parameter. If the paramteter is mandatory, you must assign it a value before any program initiation. The prompt, in this case, begins with the character, M. For example,

M KEY PRINT = FALSE?

The prompts begin with the character, E, if an error is detected as in the following example:

E INT SAMPLES=-512?

Each parameter is assigned a name by the program. The name always appears in the prompt for the particular parameter. If the parameter accepts more than one value, the number of values is enclosed in parentheses following the parameter name. For example,

INT LINES(8)=?

The following table lists the various parameter types and the user responses that are allowed:

Parameter type	Prompt	Valid response
Integer	INT	-32768 -- +32767
Real	FLT	Any real number
Logical	KEY	T[RUE] or F[ALSE]
Character	CHR*n (where n is a numeric character)	A character string less than n characters long

If the parameter is of the type character, the string length is indicated in the prompt. For example,

CHR*3 NAME=?

The string length is the maximum number of characters allowed in the string. Note that the prompt may contain a value following the sign, =, denoting the current value assigned to the parameter. For example,

INT SAMPLES=512?

You can either accept this value by pressing the carriage return key or enter new value in response to the prompt.

6.1.2 FUNCTION MONITOR

The Function Monitor (FMON) accepts the processing task from the Command Interpreter if the specified function is a central processor function. FMON loads and executes the desired function and maintains two separate tables for the currently loaded functions and the functions that are considered important enough to be loaded at all times. In the table of currently loaded functions, an upper limit is set to the number of functions that can reside in the table at the same time. This limit is normally four. The least used function is unloaded if at any instant the table is full. FMON returns the control to the Command Interpreter following the execution of the command.

6.1.3 DISPLAY MONITOR

If an IPOS command specifies a display processor function, the Command Interpreter transfers control to the Display Monitor (DMON). DMON functions in the same way as FMON.

In addition to its major components, IPOS uses the following disc-based subsystems to execute the image application programs:

6.1.4 IMAGE I/O SYSTEM

The Image I/O system handles all the input and output operations related to the image data sets. Whenever an application program tries to access an input image, it activates the Image I/O system. The Image I/O system opens and closes the image files and performs the necessary conversions on these files. Since Image I/O system manages all the input and output operations in the image data, your program can access different data in a standard and consistent manner. The actual coding of your program is relatively independent of the physical aspect and location of the image data.

6.1.5 FUNCTION LIBRARY

The Function Library maintains all the functions used by IPOS. These functions are identified by their names that can be up to 15 characters (alphanumeric or single quotes) long and must begin with a letter. You may assign more than one name to the same function. Spaces between two characters are not allowed in the name.

6.1.6 IMAGE DIRECTORY

Each user is provided with a directory, the Image Directory, that maintains all the images the user can access. The names of the images can be up to 23 characters (alphanumeric or single quotes) long and must begin with a letter. No spaces are

allowed in the names. In the event of a system crash all your files are protected in the directory.

6.1.7 TAPE DIRECTORY

Each user is provided with a tape directory which provides an offline database system. The CPU OFFLINE and the system level ONLINE functions are used to store user images on tape and retrieve them later. An entry is made in the tape directory containing the name of each image and the tape number of the offline tape.

7.0 INTERNATIONAL IMAGING SYSTEMS (I^2S) SYSTEM PROCESSOR FUNCTIONS

The I^2S Image Processing Operating System (IPOS) is a special purpose disc-based software system which controls image processing programs. This system operates under the control of the host operating system. A command interpreter functions as the main control program for IPOS. This interpreter prompts the user for input commands, passes the command then links to the appropriate system procedure to execute the specific command. These commands are categorized as System Processor Functions, CPU Processor Functions or *Display Processor Functions*. The System Processor is invoked when the utility module name is preceded by the "\" character.

7.1 SYSTEM UTILITY MODULES

The following paragraphs describe the utilities invoked by the IPOS System Processor.

7.1.1 CANCEL

Allows user to abnormally terminate any central processor function program that is currently running. This command clears the central processor function stack.

7.1.2 DELETE

Allows user to delete one or more images from the user image file directory.

7.1.3 END

Allows user to terminate his/her image processing operation in an orderly manner. This command directs IPOS to close all image file sets, release system resources and return control to the host operating system.

7.1.4 FORGET

Delete the names of image files from the user's offline image file directory. These image files reside on magnetic tape but the image file directory is on the disc.

7.1.5 HISTORY

Allows user to retrieve and display all the command(s) executed in the creation of the particular image(s). User may regenerate the subject image(s) from the original image(s) by following the outcome of the HISTORY command in any given situation.

7.1.6 LISTCAT

Allows the user to obtain listings of the names of all images in the on-line image file directory. A complete description of specific images may also be obtained.

7.1.7 NOQUEUE

Turns off multi-queuing of SYSTEM, CPU, and DISPLAY commands. Only one function will be executed at a time, whether it is a system, cpu, or display function. This function is helpful when debugging or using the "\USE" function.

7.1.8 ONLINE

With this command the user can retrieve and image file from magnetic tape and c py it to disc. IPOS provides an offline database system using the CPU OFFLINE function and the system level ONLINE function. When the OFFLINE function is used the images are copied to tape and the tape is assigned a unique identification number. The user may then use the ONLINE function to retrieve the image from the offline tape. The image remains in the offline directory until deleted with the FORGET function. Note, that the input image name must not exist in the online tape directory.

7.1.9 QUEUE

Turns on IPOS multi-queuing of SYSTEM, CPU, and DISPLAY commands. Thus a CPU and DISPLAY function can be executed at the same time. This function is normally in effect in the System, unless the "\NOQUEUE" function has been invoked.

7.1.10 RENAME

Changes the name of an image file to a new name as specified by the user.

7.1.11 SESSION

Lists all or a portion of the user-system dialogue on the standard output device. The listing does not include the output from the HELP and LISTCAT commands.

7.1.12 TIME

User can determine the time taken by any central processor or display processor command. The user can even determine retrospectively the time taken by a command typed previously but which has not completed its execution. The TIME function, if repeated, nullifies itself, meaning neither the CPU time nor the elapsed time is listed on the output device until the TIME command is repeated again.

7.1.13 USE

This function allows the user to run IPOS functions from a disc file. After executing the commands in the disc file, the function leaves the user in interactive mode in the System. It is best to run this function with the function >NOQUEUE in effect.

8.0 INTERNATIONAL IMAGING SYSTEMS (I^2S) CENTRAL PROCESSOR FUNCTIONS

The CPU Processor is invoked when the module name is preceded by the ">" character.

8.1 CPU IMAGE INPUT/OUTPUT MODULES

The following paragraphs describe the CPU modules that are used for input/output of image files.

8.1.1 AVHRR

CCT tapes obtained from EDIS containing NOAA6, NOAA7 and TIROS-N satellite image data are read by this program and stored in image files on disc. The program provides the user with options to select any combination of 1 to 5 image data channels, select an image subsection, calibrate/not calibrate the data as it is input and stored, and create control points file.

8.1.2 COPY

This module copies from an image file, file can be subsectioned via parameters, to a new image file.

8.1.3 CONTOUR

This function produces a contoured image. The user may specify the contour interval and the value of the first and last

contours. Breakpoints are computed and contours are drawn wherever a value on the high side of a breakpoint is adjacent to one below it. Contour lines may be of uniform brightness, or optionally, the brightness of each contour may increase as the associated breakpoint value increases. The user may also specify whether the contours are to be drawn on a background of the original image or on a solid background. The user may specify the brightness of the contours if they are to be of uniform brightness and the brightness of the background if it is to be solid. With the PLATEAU option, the area between each contour may be drawn to a uniform brightness, that brightness increasing with the value.

CONTOUR accepts either one or two inputs. If one input, it must be a single band file, which will be used as data to be contoured and will also be used as the background if desired. If there are two inputs, the first must be single band and will be used as the data to be contoured. The second will be used as the background image; it must be an image file but may be multiple band. If there are two inputs the user will not be given the option of excluding the background or of shading between the contours.

8.1.4 CZCS'ENTER

This program is designed to input CZCS images from magnetic tape. Two files are provided as outputs from this function. The first output is the image file containing up to six bands of data. The second output is a datafile containing the CRTT Documentation Record. The program can input any or all of the three images available on a single CZCS image tape. The program contains the option of extracting and printing a set of anchor points from the image. For further information on the data format of the CZCS tape and the CRTT Documentation record refer to the "Nimbus Observation Processing System (NOPS) Tape Specification T744041 CZCS CRT Tape" document available from NOAA in Washington, D.C. Note that this function is valid only for data tapes obtained from NOAA.

8.1.5 DAEDALUS

This function allows the user to read in standard format Daedalus tapes containing imagery from the Daedalus 12-channel airborne scanner. It will read the header from the tape, print pertinent information on the terminal and guarantee that the user requests only those bands present on the tape. The user is also given the option of subsectioning or subsampling the input data.

8.1.6 ELLIPSE

Generates an image whose pixel values are defined by the following function:

P(X,Y) = A + B * SQRT(kx *(X-SAMCNTR)**2 + ky * (y-LINECNTR)**2).

Allows user to generate an elliptical shaped image. The parameters kx and ky are determined by the eccentric parameter that is the eccentricity of the ellipse. The values of the pixel positions X and Y are scaled so that the horizontal or vertical position farthest from the center has a magnitude of 0.50.

8.1.7 ENTER

Enter an image from magnetic tape. Tape records cannot be longer than 2048 words for this function. Longer records can be input by breaking the records into multiple records 2048 words in length.

8.1.8 FUCINO

This function was written to fill a customer's need to read LANDSAT CCT's from Fucino. It has subsequently been found to work on images from receiving stations in Australia and South Africa. The function will read any part or all of the CCT and store it on disc. No SIAT file is produced because this data is not on the tape.

Some of the first images encountered in testing this function exhibited an apparent attempt to correct the skewing of the image due to earth rotation by offsetting the image data in the tape records. This approximates a deskew but it does not apparently correct for other image distortions, and it precludes the use of the System geometric correction functions and map projections. Therefore, the function was written to scan for such data offsets and remove this deskewing, effectively "reskewing" the data. The output images can then be input to any of the System functions that correct LANDSAT distortions and warp images.

8.1.9 GENERATE

Generates an image whose pixel values are defined by the following function:

P(X,Y,Z,T) = A * X + B * Y + C * Z + D * T + E,

where X = (SAMPLE LOCATION OF P)/(no. of Samples-1)
Y = (Line Location of P)/(No. of Lines-1)
Z = (Spectral Location of P)/(No. of Spectral bands)
T = (Temporal Location of P)/(No. of Temporal bands)

A,B,C,D, and E are coefficients defined by the IMAGCOEF parameter.

8.1.10 INPUT'IMAGE

This function allows a user to interactively make an image, or add to an existing image that was previously made by ">INPUT'IMAGE." The user defines the datatype and size of the

image he/she wishes to make, and gives the location within the image to begin filling with pixel values. If there is an input image, the size of the output image will be the same as the input's. The datatype will be the same, by default. The organization of the output will be SLB. The input image will be copied to the output image up to the starting location specified by the user, then the output will continue to be filled by VALUES. The user may input pixel values as individual values, or in run-length encoded form. The function will continually prompt for pixel values until the user inputs only a carraige return. This way, the user doesn't have to type in all values at once. If the user wishes to quit and the image is not completely filled, the function will print out the sample, band, and line locations of where in the image the user left off, so he/she can resume making the image at a later date.

8.1.11 LANDSAT

Input a Landsat MSS scene from magnetic tape. The tape must be in the standard NASA format of 4 image files plus the optional fiftn file that contains the platform orientation data. The user may select any subsampled scene, or a subsection 512 x 512 full resolution scene.

8.1.12 LANNION

CCT tapes obtained from Lannion, France containing NOAA6 and NOAA7 satellite image data are read by this program and stored in image files on disc. This program also provides the user with options to select any combinations of 1 to 5 image data channels, select image subsection and calibrate/not calibrate the data as it is input and stored.

8.1.13 LCOPY

Module to copy image and control point files and perform a linearization algorithm on the image and control point data. See 5.1.7 for detail description.

8.1.14 MCOPY

Module to copy and scale a control point file.

8.1.15 NEWLAND

The EROS Data Center designed a new LANDSAT CCT format and implemented it during 1980. This format eliminates many of the problems associated with the previous format but consequently it is not compatible with existing tape reading programs. This function was designed to handle the new format in all of its varied forms. The new LANDSAT CCTs will contain both RBV and MSS data in both a partially processed and a fully processed form using either a band-interleaved or a band-sequential data organization.

Multiple tape densities will also be available. The NEWLAND function will handle all of these possibilities. It will also print a block of directory information on the terminal and store either of the annotation or ancillary records in separate disk files for later use. The function can support subsectioning or subsampling pling of the tape data.

8.1.16 ORBIT'CNPT

This program was derived from some basic subroutines of the ORBIT program. The program is used to generate control point files for use in image registration. Control points can be obtained or NOAA6, NOAA7, TIROS-N and NIMBUS-7 (CZCS). The EDIS CCT tapes and respective ephemeris file are inputs to this program.

8.1.17 POLYGON'ENTER

This function is designed to allow users to input polygon data into the system. It conforms to the "Experimental Cartography Unit's (ECU) Format for the Exchange of Cartographic Data on Magnetic Tape" as used in Great Britain. However, the user must know, in advance, how many different polygon typ , are to be read and the maximum number and size of distinct pol gon regions that any type may have. This corresponds to the System 101 concepts of "class" and "region." For example, forest might be a polygon type of "class" while a specific grove of trees would represent a distinct "region." Before reading in a tape of landcover types, the user would be expected to know the exact number of different landcover types or "classes," the maximum number of different polygons or "regions" that have the same landcover type, and the maximum number of points that were used to outline the largest or most complex polygon on the tape. These three parameters are necessary to open the disc file and allocate space, and so must be known in advance.

Since the ECU format allows for multiple polygon identifiers, the program allows the user to select which identifier to key on. This is called the "feature code," and it is usually 6 characters long. If the last three characters in this field are a number less than 256, then they will be stored in the file and used later as the brightness value when plotting he polygons in the display. They can then be converted to color or other intensities using display functions such as ^COLORS and ^PLIM. If any of the encountered polygons have feature codes outside the range of 0-255, then all the polygons will be labeled by number, from 1 to NPOLYGONTYPES.

The function also makes provision for island polygons and nested polygons by optionally storing a "nesting code" from another feature code field. This nesting code will be zero when there are no constraints on the polygon, i.e. it is not an island inside some larger polygon. The code is incremented each time that polygon is found to be an island inside a larger polygon.

Therefore, a "3" nesting code polygon, which can only exist if it is entirely surrounded by a "2" nesting code polygon, which can only exist if surrounded by a "1" polygon, and so forth. Technically, functions that plot polygons on the display are coded to plot polygons in order of nesting code, so that high nesting codes are plotted last and thereby correctly overwrite their surroundings.

The user can optionally input control points that define a transform between the coordinate system of the polygons and a registered image coordinate system. If no control points are provided then the polygon coordinates are stored as is. If any control points are provided, then three or more must be provided. Typically, they will be corner points. The program is limited to eight control point pairs and the transform they define should not be too complex or errors can result. If three to five pairs are provided a first order transform will be used. If six to eight points are provided the function will use a second order transform. The user is warned to avoid colinear control points as they can lead to nonconvergent approximations in calculating the transform.

8.1.18 SCAN

This command allows the user to create disc images from the Optronics C-4500. The user may create black and white or color images of any size up to 8192 lines by 8192 samples. The data is normally swapped left to right so that the resultant image will not be a mirror copy. The image may also be negated before being written to disc.

8.1.19 SEISMIC

This function will allow the user to enter seismic profiles from a CGG seismic data tape as images for processing by the System 101.

8.1.20 TD'FIND

This function is similar to the function >FIND. It allows the user to quickly find where a given map location on the table digitizer is in an image. It requires the input of a transform file from the function ^TD'INIT.

8.1.21 TD'TRAIN

This function emulates ^TRAIN except that training areas are defined using the table digitizer on a map. A display is therefore not required. However, the function must be provided with a transform file from the function ^TD'INIT. This allows the table digitizer coordinates to be mapped onto the image coordinate system. Otherwise the function is identical to 'TRAIN.

8.1.22 TRANSFER

Output an image to magnetic tape. The image names, sizes, datatypes and organizations are printed on the line printer at the end of the transfer.

8.2 CPU IMAGE STATISTICAL MODULES

The following paragraphs describe the CPU modules that are used to perform statistics on image files.

8.2.1 CORRELATION

Computes and outputs to the user the minimum, maximum, mean, standard deviation and correlation matrix over either the spectral or the temporal dimension.

8.2.2 HECTARE

HECTARE computes the number of hectares in an image for each class in a supervised or unsupervised classification image. A table containing the class number, number of hectares per class, Percent coverage and the number of pixels is output to the session device and optionally, the line printer (BOTH=TRUE). For supervised classification the class names as well as the class number are printed in the table. The number of square meters per hectare should be checked in relation to deskewed or nondeskewed Landsat images.

8.2.3 HISTOGRAM

Compute the PDF histogram (and, if desired, the cumulative density function histogram) of an image.

8.2.4 MINMAX

Determine and output to the terminal for each spectral band and for each temporal band:

1. Sample number
2. Line number
3. Min value
4. Max value

8.2.5 P'PREPARE

PRINT'PREPARE accepts as input, statistics files created by the functions >SEPARATE and >SLOWPREPARE.

For a >SEPARATE input, it will allow the analyst to list to the line printer or both the line printer and the session device: Covariance, Mean, Min, Max, Standard Deviations, and the Covariance Inverse.

For a >SLOWPREPARE input, it will allow the analyst to list: V matrix used in the quadratic computations, Mean, Min, Max, and the Standard Deviations of each class, and the Confusion and Bilateral Bias Matrices.

The Confusion matrix is a matrix indicating the spectral confusion among the paradigms of all classes (taken two at a time) under consideration.

The Bilateral Bias Matrix is the matrix of constants used to adjust or bias a quadratic being evaluated.

Additionally, a training legend indicating the name of the class(es), the number of points in the training set and the Determinant of the Covariance, is listed.

If the default parameters are used, everything possible is listed.

The number of classes (in the case of the >SLOWPREPARE input) must be less than 65.

The type and subtype of all inputs must be the same.

8.2.6 P'STATISTICS

This function prints out the statistics and eigen information stored in files created by the >STATISTICS function.

8.2.7 PROBIT

Given two thematic classification maps over the same geographical region, PROBIT shows in histogram format how all the pixels of class i of the first class map are reclassified in the second class map, for i=0,1,,, NCLASS. The output listed to the session device includes the original class, the total number of points in the original class and for each new class ID, the number of pixels and the fraction of the total.

8.2.8 STATISTICS

This function computes the MEAN, MIN, MAX and standard deviation vectors, as well as the covariance matrix, eigenvalues and eigenvectors of an image using the host system CPU. The results can be used as an input to the KL transform functions or to direct the specification of other transforms.

8.3 CPU MULTI-IMAGE MODULES

The following paragraphs describe the CPU modules that are used manipulate Multi-image files.

8.3.1 MOSAIC

This module will generate a composite image from spatially adjoining subimages.

8.3.2 INSERT

The first image is inserted into the second image in the spatial dimensions (samples or X and lines or Y). All other dimensions (spectral, temporal, etc.) must match.

8.3.3 UNITE

Creates a multiband or multitemporal image from any number of input images. The organization of the output data set may be user specified. The size of all dimensions other than the dimension to be united must be equal across all input images.

8.4 CPU HARDCOPY GENERATION MODULES

The following paragraphs describe the CPU modules that are used to obtain hardcopy output.

8.4.1 APPLICON

This function generates Applicon color plotter tapes from System images. It accepts either three band or one band images, producing color or B/W output, respectively. An optional STASH file can be supplied to restore color adjustments.

8.4.2 LPPLOT

Generates an 8-grey level plot of input image on terminal, line printer, or both.

8.4.3 PPLOT

Generates a symbol plot from the input image on the line printer, using a user specified symbol table.

8.4.4 PRINT

Outputs an image-type data set to the terminal (default), the line printer or both. Used to obtain a spatial representation of ac brightness values.

8.4.5 RECORD

This command provides a means of recording disc images on film using a Optronics C-4500. The input image may be any size up to 8192 lines by 8192 samples. The user specifies the region on the film to be recorded and any scaling of the data which may

be required. Three band inputs will be recorded in color, one band inputs in black and white.

8.4.6 VPLOT

Produces simulated image output on an appropriate Versatec printer/plotter using a random dot pattern. The scale of the output image can be controlled by specifying the size of pixels in Versatec dots. The software assumes 200 dots per inch with an overall size of 2032 dots in the samples dimension.

8.5 CPU IMAGE ARITHMETIC MODULES

The following paragraphs describe the CPU modules used to perform arithmetic operations on images.

8.5.1 ABSOLUTE'VALUE

Generates an output image for each input image. The elements of the output image are the absolute values of the corresponding input elements.

8.5.2 ADD

Computes the weighted sum of n input images to produce a single output image. The weighted sum is performed on a pixel-by-pixel basis for each band or channel.

8.5.3 DIVIDE

Compute the weighted quotient of two images on a pixel-by-pixel basis. The program divides the first image by the second image. The user may specify weighting factors if desired. Used to divide one band (image) by another to form a ratio image.

8.5.4 MULTIPLY

Computes the weighted product of input images on a pixel-by-pixel basis and outputs a new image.

8.6 CPU IMAGE MASKING/BOOLEAN MODULES

The following paragraphs describe the CPU modules used in performing masking and Boolean operations on images.

8.6.1 CAND

CAND operates on two input classification maps, producing as a consequence, a new thematic map. Basically, it allows the analyst to detect stable regions, monitor the effect of different decision rules during the classification analysis, and to perform an analysis of nonreject regions.

CAND requires two thematic maps or masks as inputs and generates an output thematic map according to the following rule:

If the first data set is a picture or classification output and the second is a mask, the output data set is similar to the first data set except that pixels not covered by the mask are zeroed out.

8.6.2 COR

COR allows the user to combine the results of two classifications into a single output class map. It accepts two input class maps in an order such that the first map is considered dominant. The function will then "OR" these two maps according to the following decision theory:

```
IF DIFFERENT'CLASSES = FALSE AND

          IF INPUT1 < > 0 THEN OUT = INPUT1
          IF INPUT1  =  0 THEN OUT = INPUT2

IF DIFFERENT'CLASSES = TRUE AND

          IF INPUT1 < > 0 THEN OUT = INPUT1
          IF INPUT1  =  0 THEN OUT = INPUT2 + OFFSET
```

In other words, the first class map is filled in by the second class map where the pixel in the first map was a reject and is now equal to zero. The second class map values can be added with an offset to distinguish them from the corresponding class numbers in the first map. This is helpful if the class numbers in each map are not identical in meaning. The user can use this function to combine results from a hierarchical classification procedure where the most detailed classes are carefully obtained and then a more general classification is performed on the background pixels to fill in the remainder of the map. Again, the first input is the dominant one; the second input will only appear where the first has no classes.

8.7 CPU RADIOMETRIC TRANSFORMATION MODULES

Radiometric transformation operations are performed on image by the modules described in the following paragraphs.

8.7.1 DESTRIPE

Used to remove the banding or striping effect present in multi-spectral Scanner images (such as Landsat images) by suppressing sudden changes in mean and variance between adjacent lines.

8.7.2 EXPONENTIAL

Performs the transformation

Y = [BASE ** BIAS * (X - OFFSET)] - 1

where X is the intensity of an input image point and Y is the corresponding output point. With the default parameter values, the transformation is Y=exp(x)-1. The (-1) term is included so that the transform exactly inverts the LOG transform that employs a (+1) additive bias.

8.7.3 HISTEQ

Perform histogram equalization (redistribution) on an image. This function uses the histogram of the image to generate a position invariant intensity mapping which will force the cumulative histogram of the output image to be linear. The effect of this operation is to increase the contrast between the more densely populated intensity levels.

8.7.4 LOGARITHM

Computes the logarithm of each pixel according to the following formula:

Y = OFFSET + LN (X + 1) / (BIAS * LN (EASE))

where X is the original pixel intensity and Y is the transformed pixel intensity. The additive constant of 1. is added to avoid computing the log of zero.

8.7.5 PIECEWISE'LIN

Applies a piecewise-linear intensity mapping to an image. The user must supply break point pairs that describe the mapping function. Each pair consists of an input level and the corresponding output level, i.e. 10, 20 maps input level 10 to output level 20. Values falling between those specified are determined by interpolation between adjacent breakpoints. Values falling above or below the endpoints are mapped to their respective endpoints.

8.7.6 SCALE

Automatically scale an image linearly into a user-defined intensity range. Optionally the user may select percentage clip levels which use the histogram of the images to determine clip levels.

8.7.7 SUN'ANGLE

Compensate for the radiometric differences between two images collected at different sun angles. The user specifies the two sun angles and the image that is to be corrected.

8.8 CPU MULTI-BAND SPECTRAL TRANSFORMATION MODULES

Radiometric transformations are performed on multi-banded spectral images by the modules described in the following paragraphs.

8.8.1 FFT1D

Performs the one dimensional Fourier transform on the spectral or temporal dimension of an image. Since this function is used primarily as a feature space rotation, the scale factor was dropped from the computation to increase the speed of the function. As a result the magnitude at any given frequency is four times that of the true Fourier frequency.

This command computes the Fourier transform of the spectral channels of the image at each pixel in the image. The user may reduce the number of features in the output image by using the parameter that specifies how many of the Fourier frequencies to output. Note that the Fourier transform is only applicable to images with spectral or temporal dimensions that are an integral power of 2.

8.8.2 HADAMARD

Performs a one-dimensional Hadamard transform on the spectral or temporal dimension of an image. The dimension to be transformed must be an integral power of 2. Since this function is used primarily as a feature space rotation, the scale factor was dropped from the computation to increase the speed of the function. As a result the magnitude at any given sequence is twice that of the true Hadamard sequence.

This command computes the Hadamard transform of the spectral channels of the image FOREST at each pixel in the image. The user may reduce the number of features in the output image by using the parameter which specifies how many of the Hadamard sequences to output. Note that the Hadamard transform is only applicable to images whose spectral or temporal dimensions are an integral power of 2.

8.8.3 KL'TRANSFORM

This function performs a Karhunen-Loeve transform on an image in the host system CPU. The image statistics file must be

supplied and can be made using ½STATISTICS. The form of Karnunen-Loeve transform used is:

```
                 NBANDS
                 ---
                 \
OUT(comp) =       >  EIGEN(band,comp) * [ IN(band) - MEAN(band) ]
                 /
                 ---
                 band = 1
```

This transform is equivalent to a principal components, feature space rotation of an image. Each component will contain a percentage of the overall variance equivalent to its eigenvalue in the statistics file. The output cannot have more components than the input has bands, but the user can specify fewer than the maximum number of components. The output file is real unless "BYTEOUTPUT" has been specified, in which case the result of the transform is scaled to a range of 0-255 by assuming the theoretical maximum and minimum the transform can produce and dividing accordingly. The user should note that this scaling can lead to a loss of precision because the theoretical limits are usually not reached in the output.

8.8.4 NORMALIZE

Divides each spectral band by the sum of the spectral bands. An option is provided to divide each band by a weighted sum of all the bands.

8.8.5 RATIO

Performs one of two spectral ratio operations to all bands of the input image, the options are

```
(1) S(i+1) / S(i),                   for i = 1,NBANDS-1
(2) [S(i+1)-S(i)]/[s(i+1)+s(i)],     for i = 1,NBANDS-1
```

where S is a vector of length NBANDS. The ratios are calculated for each pixel in the image.

8.8.6 STANDARDIZE

Standardize the spectral or temporal dimension according to the following change of variables: Y(I)=(X(I)-M(I))/SIGMA(I), where X(I) is the pixel value, M(I) is the estimated mean, SIGMA(I) is the estimated standard deviation.

8.9 CPU SPATIAL TRANSFORMATION MODULES

The modules used to perform spatial manipulation of images are described in the following paragraphs.

8.9.1 CONVOLVE

This function performs a two-dimensional spatial convolution on a one band display image utilizing up to a 256 element kernel specified by the user. The algorithem utilized is of the following form:

$$p'(x,y) \quad \sum_{i=-m/2}^{m/2} \; \sum_{j=-n/2}^{n/2} p(x+i),y+j) * weights(i,j)$$

where P'(x,y) is resultant pixel intensity after convolution x is the line position of the pixel, y is the line number of the pixel, P(x,y) is the original pixel intensity, WEIGHTS is an n by m array containing the convolution kernel, n is the number of kernel elements per row, and m is the number of kernel elements per column.

8.9.2 FFT2D

Perform the two-dimensional discrete Fourier transform on a real-valued image.

This command computes the two dimensional Discrete Fourier transform of a section of the image. Note that the number of lines and number of samples must both be an integral power of 2 for this function. The FFT function will not handle an image larger than 512 by 512.

8.9.3 IFT2D

Performs the inverse two-dimensional Fourier transform of a data set of the form produced by FFT2D.

This command computes the inverse two-dimensional Discrete Fourier transform of the transform image X'DOWNTOWN. The input image must be in a format compatible with the output of the FFT2D function.

8.10 CPU SPATIAL FILTER MODULES

Spatial filtering of images is performed by the modules described in the following paragraphs.

8.10.1 EXPFLT

Generates and applies an exponential shaped frequency domain filter to a complex type image previously Fourier transformed by FFT. The output from "EXPFLT" must be inverse Fourier

transformed in order to return the image to the spatial domain. If no input file is provided, the output file will contain a complex filter data set that can be subsequently used to multiply by FFT'd images. This saves time by eliminating the recomputing of the filter for each filtering operation if the saved filter is to be used many times. The form of the filter is:

H(F) = A + B * EXP (C * RADIUS)

where A, B and C are user specified coefficients, and RADIUS is the radial distance from the center of the complex Fourier domain. The radius is measured in cycles per pixel so that the largest frequency in either the horizontal or vertical direction is 0.5 or -0.5 cycles per pixel.

8.10.2 FADE'FILTER

Computes and either saves (or applies) a cosine fadeout mask to image data. The mask value for any given pixel is a function of the spatial coordinates of the pixel which, when multiplied into the data, causes pixels in the spatial boundaries of the image to fade to zero.

After computing the mask value for each pixel the function applies the multiplicative weight GAIN either to the mask or the masked data and adds a constant so that either

CASE 1 (mask only): MASK'(S,L) = MASK(S,L) * GAIN + DC
CASE 2 (apply mask): FADED'IMAGE(S,L) = INPUT'IMAGE (S,L) * MASK (S,L)*GAIN + DC

The function is typically used

- prior to a convolution process in order to suppress edge-ring effects caused by a discontinuous boundary.
- prior to the mosaic process in order to create a boundary of zeros.

8.10.3 GAUSS'FILTER

Generates and applies a Gaussian shaped frequency domain filter to a complex type image previously Fourier transformed by FFT. The output from "GAUSS'FILTER" must be inverse Fourier transformed in order to return the image to the spatial domain. If no input file is provided, the output file will contain a complex filter data set which can be used subsequently to multiply by FFT'd images. This saves time by eliminating the recomputing of the filter for each filtering operation if the save filter is to be used many times.

8.11 CPU GEOMETRIC TRANSFORMATION MODULES

The modules used to perform geometric transformations on images are described in the following paragraphs.

8.11.1 ALBERS

Maps a Landsat image into the Albers Conic Equal Area Projection. All known systematic distortions inherent in the imagery are compensated for (including mirror velocity profile). If the Scene Image Annotation Tape (SIAT) file is available (present on recent CCT's, and generated by the program LANDSAT), compensation for platform pitch, roll, yaw, altitude fluctuations, etc. is also provided. The user may also input empirically determined control point data to refine the correction.

Either one or two input files may be specified. If one file is input, it is assumed to be the image to be corrected. If two files are input, one must be a Landsat image, and the other the corresponding SIAT file (optionally generated by the program "LANDSAT").

If two output files are designated by the user, the first file will contain the corrected image and the second file will contain a ACEA grid which can be overlayed using the "WRITE-GRAPHICS" display function.

If a SIAT-type input file is provided, none of the mandatory parameters described below are prompted for.

Input all angles in the following real format: (+/-)ddd.mmsss To input the longitude W117 degrees 52 minutes 28.46 seconds, the user should input: -117.522846. Input all points in the southern or western hemispheres as negative numbers, rest are positive.

8.11.2 AVERAGE

Generates an image of reduced size by partitioning the input image into N sample by M line blocks, and averaging the pixel values within each block to calculate the corresponding pixel values in the output image.

AVERAGE compresses the image size by a factor of two in each dimension. Each output point represents the average of four input points.

8.11.3 DESKEW

Geometrically corrects a LANDSAT image to remove the distortions resulting from the rotation of the earth relative to the satellite during the exposure, and the 3240 to 2340 aspect ratio

inherent in all LANDSAT data. The image may also be rotated to true North during this operation if desired.

8.11.4 FASTMAGNIFY

Magnify an image by different integer scale factors in the samples and lines direction. The magnification is accomplished by repeating pixels.

8.11.5 FRAME

Corrects for the geometrical distortions inherent in a central projection (frame) image recording device (i.e. conventional camera or TV). The user may input attitude parameters that define the viewing geometry, control points, or both in order to specify the transformation.

The control points may be typed in manually, or provided in the form of a control point file generated by program CNPT.

8.11.6 KRUEGER

Maps a Landsat image into the GAUSS'KRUEGER Map Projection. All known systematic distortions inherent in the imagery are compensated for (including mirror velocity profile). If the Scene Image Annotation Tape (SIAT) file is available (present on recent CCT'S, and generated by the program LANDSAT), compensation for platform pitch, roll, yaw, altitude fluctuations, etc. is also provided. The user may also input empirically determined control point data to refine the correction. The LANDSAT scene to be processed must be entirely contained within a 6 degree (in longitude) GAUSS'KRUEGER zone. If an image straddles two zones, it must be partitioned into separate subimages for processing. The user is notified by an error message if this is the case.

Either one or two input files may be specified. If one file is input, this is assumed to be the image to be corrected. If two files are input, one must be a landsat image, and the other the corresponding SIAT file (optionally generated by the program "LANDSAT").

If two output files are designated by the user, the first file will contain the corrected image and the second file will contain a GAUSS grid (bit-type) that can be overlayed using the "WRITEGRAPHICS" display function.

If a SIAT-type input file is provided, none of the mandatory parameters described below are prompted for.

Input all angles in the following real format: (+/-)ddd.mmsss To input the longitude W117 degrees 52 minutes 28.46 seconds, the user should input: -117.522846. Input all

points in the southern or western nemispheres as negative numbers, rest are positive.

8.11.7 LAMBERT

Maps a Landsat image into the Lambert Conformal Conic Projection. All known systematic distortions inherent in the imagery are compensated for (including mirror velocity profile). If the Scene Image Annotation Tape (SIAT) file is available (present on recent CCT's, and generated by the program LANDSAT), compensation for platform pitch, roll, yaw, altitude fluctuations, etc. is also provided. The user may also input empirically determined control point data to refine the correction.

Either one or two input files may be specified. If one file is input, it is assumed to be the image to be corrected. If two files are input, one must be a Landsat image, and the other the corresponding SIAT file (optionally generated by the program "LANDSAT").

If two output files are designated by the user, the first file will contain the corrected image and the second file will contain a LCC grid (bit-type) which can be overlayed using the "WRITEGRAPHICS" display function.

If a SIAT-type input file is provided, none of the mandatory parameters described below are prompted for.

Input all angles in the *following real format:* (+/-)ddd.mmsss To input the longitude W117 degrees 52 minutes 28.46 seconds, the user should input: -117.522846. Input all points in the southern or western hemispheres as negative numbers, rest are positive.

8.11.8 LAT'LONG

Maps a Landsat image into a Latitude-Longitude Projection. All known systematic distortions inherent in the imagery are compensated for (including mirror velocity profile). If the Scene Image Annotation Tape (SIAT) file is available (present on recent CCT's, and generated by the program LANDSAT), compensation for platform pitch, roll, yaw, altitude fluctuations, etc. is also provided. The user may also input empirically determined control point data to refine the correction.

Either one or two input files may be specified. If one file is input, this is assumed to be the image to be corrected. If two files are input, one must be a Landsat image, and the other the corresponding SIAT file (optionally generated by the program "LANDSAT").

If two output files are designated by the user, the first file will contain the corrected image and the second file

will contain a UTM grid (bit-type) that can be overlayed using the "WRITEGRAPHICS" display function.

If a SIAT-type input file is provided, there is no prompting for the parameters described below.

Input all angles in the following real format: (+/-)ddd.mmsss To input the longitude W117 degrees 52 minutes 28.46 seconds, the user should input: -117.522846. Input all points in the southern or western hemispheres as negative numbers, rest are positive.

8.11.9 MAGNIFY

MAGNIFY (or contract) an image by different scale factors in the samples and lines direction. The scale factors may be non-integer (for integer scale factors "FASTMAG" should be used).

8.11.10 NAPOLYCONIC

Maps a Landsat image into the North American Polyconic Projection. All known systematic distortions inherent in the imagery are compensated for (including mirror velocity profile). If the Scene Image Annotation Tape (SIAT) file is available (present on recent CCT's, and generated by the program LANDSAT), compensation for platform pitch, roll, yaw, altitude fluctuations, etc.,is also provided. The user may also input empirically determined control point data to refine the correction.

Either one or two input files may be specified. If one file is input, it is assumed to be the image to be corrected. If two files are input, one must be a Landsat image, and the other the corresponding SIAT file (optionally generated by the program "LANDSAT").

If two output files are designated by the user, the first file will contain the corrected image and the second file will contain a NAPC grid (bit-type) which can be overlayed using the "WRITEGRAPHICS" display function.

If a SIAT-type input file is provided, none of the mandatory parameters described below are prompted for.

Input all angles in the following real format: (+/-)ddd.mmsss To input the longitude W117 degrees 52 minutes 28.46 seconds, the user should input: -117.522846. Input all points in the southern or western hemispheres as negative numbers, rest are positive.

8.11.11 POLAR

Maps a Landsat image into the Universal Polar Stereographic Projection. All known systematic distortions inherent in

the imagery are *compensated for* (including mirror velocity profile). If the Scene Image Annotation Tape (SIAT) file is available (present on recent CCT's, and generated by the program LANDSAT), compensation for platform pitch, roll, yaw, altitude fluctuations, etc., is also provided. The user may also input empirically determined control point data to refine the correction.

Either one or two input files may be specified. If one file is input, it is assumed to be the image to be corrected. If two files are input, one must be a Landsat image, and the other the corresponding SIAT file (optionally generated by the program "LANDSAT").

If two output files are designated by the user, the first file will contain the corrected image and the second file will contain a UPS grid (bit-type) which can be overlayed using the "WRITEGRAPHICS" display function.

If a SIAT-type input file is provided, none of the mandatory parameters described below are prompted for.

Input all angles in the following real format: (+/-)ddd.mmsss To input the longitude W117 degrees 52 minutes 28.46 seconds, the user should input: -117.522846. Input all points in the southern or western hemispheres as negative numbers, rest are positive.

8.11.12 ROTATE

Rotate an image by an arbitrary angle in the plane of the image.

8.11.13 SOM

Maps a Landsat image into the Space Oblique Mercator Projection. All known systematic distortions inherent in the imagery are compensated for (including mirror velocity profile). If the Scene Image Annotation Tape (SIAT) file is available (present on recent CCT's, and generated by the program LANDSAT), compensation for platform pitch, roll, yaw, altitude fluctuations, etc., is also provided. The user may also input empirically determined control point data to refine the correction.

Either one or two input files may be specified. If one file is input, it is assumed to be the image to be corrected. If two files are input, one must be a Landsat image, and the other the corresponding SIAT file (optionally generated by the program "LANDSAT").

If two output files are designated by the user, the first file will contain the corrected image and the second file will contain a SOM grid (bit-type) which can be overlayed using the "WRITEGRAPHICS" display function.

If a SIAT-type input file is provided, none of the mandatory parameters described below are prompted for.

Input all angles in the following real format: (+/-)ddd.mmsss To input the longitude W117 degrees 52 minutes 28.46 seconds, the user should input: -117.522846. Input all points in the southern or western hemispheres as negative numbers, rest are positive.

8.11.14 STATEPLANE

Maps a Landsat image into the State Plane Coordinate Projection. All known systematic distortions inherent in the imagery are compensated for (including mirror velocity profile). If the Scene Image Annotation Tape (SIAT) file is available (present on recent CCT's, and generated by the program LANDSAT), compensation for platform pitch, roll, yaw, altitude fluctuations, etc., is also provided. The user may also input empirically determined control point data to refine the correction.

Either one or two input files may be specified. If one file is input, it is assumed to be the image to be corrected. If two files are input, one must be a Landsat image, and the other the corresponding SIAT file (optionally generated by the program "LANDSAT").

If two output files are designated by the user, the first file will contain the corrected image and the second file will contain an SP grid (bit-type) which can be overlayed using the "WRITEGRAPHICS" display function.

If a SIAT-type input file is provided, none of the mandatory parameters described below are prompted for.

Input all angles in the following real format: (+/-)ddd.mmsss To input the longitude W117 degrees 52 minutes 28.46 seconds, the user should input: -117.522846. Input all points in the southern or western hemispheres as negative numbers, rest are positive.

8.11.15 UTM

Maps a Landsat image into the Universal Transverse Mercator Projection. All known systematic distortions inherent in the imagery are compensated for (including mirror velocity profile). If the Scene Image Annotation Tape (SIAT) file is available (present on the recent CCT's, and generated by the program LANDSAT), compensation for platform pitch, roll, yaw, altitude fluctuations, etc., is also provided. The user may also input empirically determined control point data to refine the correction. The LANDSAT scene to be processed must be entirely contained within a 6° (in longitude) UTM zone. If an image straddles two zones, it must

be partitioned into separate subimages for processing. The user is notified by an error message if this is the case.

Either one or two input files may be specified. If one file is input, this is assumed to be the image to be corrected. If two files are input, one must be a Landsat image, and the other the corresponding SIAT file (optionally generated by the program "LANDSAT").

If two output files are designated by the user, the first file will contain the corrected image and the second file will contain a UTM grid (bit-type) that can be overlayed using the "WRITEGRAPHICS" display function.

If a SIAT-type input file is provided, none of the mandatory parameters described below are prompted for.

Input all angles in the following real format: (+/-)ddd.mmsss To input the longitude W117 degrees 52 minutes 28.46 seconds, the user should input: -117.522846. Input all points in the southern or western hemispheres as negative numbers, rest are postive.

8.11.16 WARP

Apply a spatial warping to an image by fitting a polynomial function to a set of User specified control points to define the transformation. The control points may be entered manually, supplied in the form of a file previously generated by use of the control point picking routine "CNPT" or "ORBIT'CNPT."

8.12 CPU CLASSIFICATION MODULES

The following paragraphs describe the modules used to perform image classifications via CPU functions.

8.12.1 CCT'CLASSIFY

CCT'CLASSIFY applies a modified maximum log likelihood rule (with thresholding) on a pixel by pixel basis along either the spectral or temporal dimension to classify a set of unknown pixels into one of N classes. Ties are resolved by selecting the candidate first examined. The class statistics are assumed to be strongly stationary and Gaussian. The input image is assumed to be on tape in the LANDSAT format as defined by NASA.

Theoretically, the decision rule is no longer necessarily Bayes optimal because of the inclusion of a rejection function.

Computational shortcuts suggested by Eppler have been implemented.

8.12.2 CCT'MINDIST

CCT'MINDIST performs a minimum distance classification on each pixel P(SAMP,LINE,*) of a LANDSAT (tape based) image according to the following rule:

CLASS (P(SAMPLE, LINE,*)) = K if and only if

1. K = MIN C (SUM L=1,NCLASS (T(L)*(P(SAMP,LINE,L)-U(C,L)))
   (minimum weighted taxi cab distance decision rule)

CLASS(P(SAMP,LINE,*)) = 0 (REJECT) otherwise.

where U(C,L) = the mean of the Lth feature of the Cth class.

where

T(L) = INTEGER (SWEIGHT(L) * MIN0(1, CONSTANT/SIGMA(L)))
       if LOGWGHTS = FALSE and UNITWEIGHTS = FALSE

T(L) = INTEGER (SWEIGHT(L)*MIN0(1,LN(CONSTANT/SIGMA(NL))))
       if LOGWGHTS = TRUE and UNITWEIGHTS = FALSE

T(L) = 1 otherwise (I.E. UNITWEIGHTS=TRUE)

The effect of the weights is to standardize the data, so that the weighted feature L is an integer estimate of the number of deviations P(SAMP,LINE,L) is away from the mean of the class being tested (in the case LOGWGHTS=FALSE) along the direction of that feature L.

The procedure is especially useful for quick look classifications of masked sections of an LANDSAT scene, especially in situations where the features are uncorrelated and classes are separable. The input image is assumed to be on tape in the LANDSAT format as defined by NASA.

8.12.3 CCT'PARALLEL

CCT'PARALLEL performs a parallelepiped classification on each pixel P(SAMP,LINE,*) of a LANDSAT (tape based) image according to the following rule:

CLASS (P(SAMPLE,LINE,*)) = K if and only if

1. K = FIRST C such that for each feature L,
   ABS (P(SAMPLE,LINE,L)-U(C,L)) < THRESH*SIGMA(C,L)

CLASS(P(SAMPLE,LINE,*)) = 0 (REJECT) otherwise.

where U(C,L) = the mean or the Lth feature of the Cth class.

SIGMA(C,L) = THE standard deviation of the Lth feature of the Cth class.

The input image is assumed to be on tape in the LANDSAT format as defined by NASA.

8.12.4 CLASSIFY

CLASSIFY applies a modified maximum log likelihood rule (with thresholding) on a pixel by pixel basis along either the spectral or temporal dimension to classify a set of unknown pixels into one of n classes. Ties are resolved by selecting the candidate first examined. The class statistics are assumed to be strongly stationary and Gaussian.

Theoretically, the decision rule is no longer necessarily Bayes optimal because of the inclusion of a rejection function.

Computational shortcuts suggested by Eppler have been implemented. CLASSIFY does not allow you to save the map of Mahalanobis distances corresponding to the thematic map. It does allow to select out only a subset of the features to be used in the quadratic computations.

8.12.5 DIVERGENCE

DIVERGENCE performs a divergence analysis to determine those NCHAN out of MXCHAN channels that either maximize or minimize the separability of a selected set of class distributions. Class distributions are assumed to be sufficiently defined by their estimated means and covariances and separability is defined in terms of one of the following functions:

1. MAX AVERAGE DIVERGENCE
2. MAX AVERAGE TRANSFORMED DIVERGENCE
3. MAX PAIRWISE DIVERGENCE
4. MIN AVERAGE DIVERGENCE
5. MIN AVERAGE TRANSFORMED DIVERGENCE
6. MIN PAIRWISE DIVERGENCE

The Program Outputs:

1. For each class, the NCHAN 'best' channels with respect to the selected divergence function.

2. The NCHAN 'best' channels over all classes with respect to the selected divergence function.

8.12.6 MAP'STATS

MAP'STATS generates a statistics file from an image and a thematic classification map corresponding to the image. The

statistics file may be used by the maximum likelihood classification algorithm.

MAP'STATS may be used to refine estimates of class statistics.

8.12.7 MINDIST

MINDIST'CLASSIFY performs a minimum distance classification on each pixel P(SAMP,LINE,*) of an image according to one of the following two rules:

CLASS (P(SAMPLE,LINE,*)) = K if and only if

1. K = MIN C (SUM L=1,NCLASS (T(L)*(P(SAMP,LINE,L)-U(C,L)))
   (minimum weighted taxi cab distance decision rule, used when keyword parameter WCITY = TRUE)

2. K = MIN C (SUM L=1,NCLASS (T(L)*(X(SAMP,LINE,L)-U(C,L))**2))
   (minimum weighted Euclidean distance decision rule, used when keyword parameter WEUCLID = TRUE)

CLASS (P (SAMP,LINE,*)) = 0 (REJECT) otherwise.

where U(C,L) = the mean of the Lth feature of the Cth class and

T(L) = INTEGER (SWEIGHT(L) * MINO (1, CONSTANT / SIGMA(L)))
       if LOGWGHTS = FALSE and UNITWEIGHTS = FALSE

T(L) = INTEGER (SWEIGHT(L) * MINO(1, LN(CONSTANT / SIGMA(NL))))
       if LOGWGHTS = TRUE and UNITWEIGHTS = FALSE

T(L) = 1 otherwise (I.E. UNITWEIGHTS=TRUE)

The effect of the weights is to normalize the data, so that the weighted feature L is an integer estimate of the number of deviations P(SAMP,LINE,L) is away from the mean of the class being tested (in the case LOGWGHTS=FALSE) along the direction of that feature L.

The procedure also allows the use of a mask so that all pixels not under the mask are automatically placed in the REJECT class.

The procedure is especially useful for quick look classifications for masked sections of an LANDSAT scene, or in situations where the features are uncorrelated.

8.12.8 PARALLEL

PARALLEL performs a parallelepiped classification on each pixel P(SAMP,LINE,*) of an image according to the following rule;

CLASS (P(SAMPLE,LINE,*)) = K if and only if

1. K = FIRST C such that for each feature L,
ABS (P(SAMPLE,LINE,L) - U(C,L)) < THRESH * SIGMA(C,L)

CLASS (P(SAMP,LINE,*)) = 0 (REJECT) otherwise

where U(C,L) = the mean of the Lth feature of the Cth class.

SIGMA(C,L) = The standard deviation of the Lth feature of the Cth class.

The INPUT-OUTPUT requirements are as follows:

INPUT 1 (required): Image to be classified (disc-based).
INPUT 2 (required): Statistics file output by the function SLOWPREPARE
OUTPUT 1 (required): THEMATIC CLASSIFICATION MAP

8.12.9 PREPARE

This is the normal function to be used when the user wants to prepare training areas for use by the classifiers in the System. This function accepts an input file of vertices that define training areas in a given image and outputs the necessary statistics for each class to a file. The vertices file is the natural output of display (^) TRAIN and contains descriptions of each class's name, and training area locations(s). The PREPARE function combines this information with a disc-resident image file, extracts the training pixels, generates summary statistics, and outputs them to a file. The function uses a Cholesky-type algorithm to produce the statistics.

8.12.10 SEPARATE

This function is used to compute training set statistics for use with supervised classifiers. The user inputs the image containing the training area and the disc file containing vertices of the training regions acquired using the display (^)TRAIN command. The output files contain the statistics of each training area and of the feature space locations of the training set paradigms, for use as input to SLOW'PREPARE or DIVERGENCE.

8.12.11 SLOW'PREPARE

SLOW'PREPARE inputs the training set statistics of the pertinent classes from SEPARATE and creates a single output data set equivalent to that produced by PREPARE for use by the System classification functions. However, it also performs a maximum likelihood classification on the pixels contained in each training set and calculates a confusion matrix. The confusion matrix is an indication of how separable the classes are under the assumptions made, and of how well the classifier will perform on

the rest of the image. This confusion matrix makes the function run slower than PREPARE, and SLOW'PREPARE requires that SEPARATE be run on the output from display TRAIN to divide each training sample into a different file. Users who simply wish to get to the classification stage quickly should not use the SEPARATE and SLOW'PREPARE functions, but rather the simpler and faster PREPARE function.

8.12.12 T'EDIT

When the user has collected some training samples using the display (^)TRAIN function and now wishes to modify them, merging some classes, deleting others, etc., the function T'EDIT should be used. This function accepts up to eight input vertice files from *TRAIN*, processes each training sample in order, and produces a single consolidated output. The user can perform one of four actions on the input training areas:

K(eep) This option merely writes the training sample's region locations to the output file unchanged.

D(elete) The user can delete any or all regions from a training sample using this option.

M(erge) This option allows the user to merge some or all regions from the present training sample to a previously output training sample. This allows the user to combine the training areas for two classes where a confusion analysis has shown they are not spectrally separable.

S(plit) With this operation the user can create two classes from the regions of a single class's training sample.

The user can assist this editing process by using the display (^)PLOT function to preview the training areas and plan which operations to perform on which classes.

8.13 CPU UTILITY MODULES

The following paragraphs describe modules used to perform CPU utility functions.

8.13.1 BELL

Send a bell tone to the user at his terminal. May be used to signal the completion of a program by placing a call to BELL in the appropriate place in the command stack.

8.13.2 CONVERT

Converts an image data set with data type INDTYPE (bit, byte integer, etc.) to an image data set with data type OUTDTYPE and saves the converted data set.

8.13.3 DELETE

This command will delete an image from the User Image Directory. Note that this function executes in the Central Processor and as such will not execute until it reaches the top of the command stack. The System Processor DELETE function, however, will delete the function immediately since the System Processor runs asynchronously to the Central Processor. Several files can be deleted at one time.

8.13.4 FIND

"FIND" is used to find the pixel within a Landsat image that corresponds to a particular set of latitude-longitude coordinates. This is useful in finding the sample-line coordinates of a region of interest in a Landsat CCT, or an image derived from a CCT.

If a SIAT-type input file is provided, none of the mandatory parameters are prompted for (except LATITUDE and LONGITUDE).

8.13.5 HELP

The HELP function is used to explain any function in the system. It is run from the system and may be used to generate listing on either the session device or on the line printer. It may also be used to find functions when the spelling is in question. Any appendix information found in the IPOS User Manual may also be obtained.

8.13.6 OFFLINE

With this command you can store images on magnetic tape for archival storage. IPOS provides an offline database system using the OFFLINE function and the SYSTEM ONLINE function. When the OFFLINE function is used the images are copied to tape and the tape is assigned a unique identification number. The user may then use the ONLINE function to retrieve the image from the offline tape. The image remains in the offline directory until deleted with the FORGET function.

8.13.7 PHASOR

Function to convert between various representations of an image in the frequency domain. The representations include the following:

a. Magnitude-phase representation
b. Complex representation
c. Real-imaginary representation

The (Magnitude,Phase) <=> Complex conversion is given by

(MAG(S,L,B),PHAZ(S,L,B)) <=> MAG(S,L,B) * COS(PHAZ(S,L,B)) +
i*MAG(S,L,B) * SIN(PHAZ(S,L,B))

The (REAL,IMAGINARY < = > COMPLEX conversion is given by

(REAL(S,L,B),IMAGINARY(S,L,B)) <=> REAL(S,L,B)+i*IMAGINARY(S,L,B)

Input/Output Requirements:

All input and output data sets are single band images. One of 4 cases is possible:

Case 1: NIDS=1 and NODS=2 and MAGPHAZ2COMPLEX=FALSE. The input, a real complex image, is converted to two images in which the first represents the real component and the second represents the imaginary component.

Case 2: NIDS=1 and NODS=2 and MAGPHAZ2COMPLEX=TRUE. The input, a complex image, is converted into 2 output images. The first image corresponds to the magnitude or gain component. The second second image corresponds to the phase component.

Case 3: NIDS=2 and NODS=1 and MAGPHAZ2COMPLEX=FALSE. The first input is considered the real component of an image in the real, imaginary representation. The second input is the imaginary component and the output is a complex image.

Case 4: NIDS=2 and NODS=1 and MAGPHAZ2COMPLEX=TRUE. The first input is considered the magnitude component in the magnitude-phase representation and the second input is the phase component. The output is a complex image.

8.13.8 TRANSPOSE

Transposes (or exchanges) any or all of the dimensions of an image. A simultaneous reorganization is also allowed (E.G. The Spectral dimension in the original image may be corresponded to the Temporal dimension in the output image.)

9.0 INTERNATIONAL IMAGING SYSTEMS (I^2S) DISPLAY PROCESSOR FUNCTIONS

The Display Processor is invoked when the module name is preceded by the "^" character.

9.1 DISPLAY IMAGE MANAGEMENT MODULES

The following paragraphs describe the Display modules that are used for managing the Model 70 display.

9.1.1 ACQUIRE

Used to acquire the display once the terminal session has been initiated. Only required when the user needs to perform display operations. The system can run CPU and system functions without acquiring the Model 70. Once the ACQUIRE function has been invoked the IPOS 101 also initializes and continues to update the System Status Monitor. If any display function aborts with a catastrophic program error the ACQUIRE function must be run again to re-acquire the display.

9.1.2 FIND'CHANNELS

FIND'CHANNELS prints the physical channels used to display the input image. If you notice hardware memory errors on the display, this function can be used to locate them.

9.1.3 GREYSCALE

Resets the input display image and loads a grey scale in a user defined rectangular portion of the image.

9.1.4 LOCK

Lock the input images to prevent the image from being written over by subsequent display commands.

9.1.5 MEMORY'TEST

Display diagnostic routine to test up to 7 patterns on user-specified Refresh Memory channels. The user must specify the number of bit-planes for each Refresh Memory.

9.1.6 RELEASE

Releases the display from control of the current session so that another user may acquire the display.

9.1.7 SELECT

Reset the display to display the original image, with the effects of all current processing deleted. Selects the input

image for display this function is used to select images already loaded in refresh memory. It may also be used to eliminate the results of a display process and return the image to its original form.

9.1.8 UNLOCK

Unlocks the specified image.

9.2 DISPLAY IMAGE INPUT/OUTPUT MODULES

Modules used to input/output image data for the Model 70 display are described in the following paragraphs.

9.2.1 AVERAGE

This function digitizes black and white or color images which have been averaged in real time. An exponential continuous average on N frames is used. The averaging function is used to compensate for vibration to the digitizer (camera) stage and noise in the digitizer.

For color images, this function must capture three separate frames corresponding to the red, green, and blue components of the image. To accomplish this, the user must use red, green and blue photographic filters since the camera input is normally a monochrome source.

The trackball button options are:

Button A) Focus/F-stop
Button B) Start average
Button C) Hold
Button D) Exit

Button "B" is used to start or resume the digitizing process. The image (or one band of the image) is fed-back to the Model 70 repeatedly. After a few seconds of averaging, a blue "snow" appears. The image or band can now be frozen with button "C" and if it is satisfactory, kept using button "D." (The blue snow effect disappears as the FRAME parameter is increased and is not visible when the image is frozen.)

Button "D" is used to store the image (or one band of the image) into refresh memory. For COLOR images, button "D" is pressed once for each lens filter which should be used in RED, GREEN and BLUE sequence.

Button "A" is used to return to "live" picture mode for repositioning the object being digitized or to set focus or F-stop.

9.2.2 CLOSETAPE

The CLOSETAPE function is used to rewind and unload an image display tape that has been used by the ^ENTER function.

After the first ^ENTER command, you must answer a tape mount request on the console. Subsequent ^ENTER commands use the same tape until the ^CLOSETAPE function is executed.

After the execution of ^ENTER, the tape file remains open until the CLOSETAPE function is executed, the display function monitor is aborted or the entire session is terminated by the END command.

9.2.3 DELETE

You can delete one or more images from the display refresh memory using this command.

9.2.4 DIGITIZE

This program digitizes film chips (negatives) from the light table digitizer, creating either a b&w or color composite image within the refresh memories. To digitize a color image the user must use either a color positive image in conjunction with optical filters to create the color, or use three color separation negatives. When digitizing a black and white image, the user pushes the B Button to digitize the image once the focus and exposure have been adjusted. After pushing the B button the digitized image will then be selected for review. If the image is satisfactory to the user then he/she pushes the B button to exit, if not the user may push the A button to repeat the digitization process. After pushing the A button the image digitizer will again be selected to allow the user to adjust the focus, exposure, position, etc. When digitizing color images an additional step is required for the green and blue images (second and third images). In order to allow for the registration of col :r separations the program displays a difference image between the live image being digitized and the previous color digitized. When using color separations, this allows the user to adjust to position and orientation of the images so that they are properly registered. This step occurs after the user has adjusted the focus and exposure for the color being digitized. As soon as the user presses the B button indicating that the exposure and focus are acceptable, the difference image is displayed. Once the user is satified that the images are registered, he again pushes the B button and the image is digitized. At this time the digitized image is then displayed and the user again has the option of proceeding by pushing the B button or returning to redo the digitization by pressing the A button. At the completion of a color digitization, the digitized color image will be displayed. In order to aid the user, the cursor is turned on in the appropriate

color during the digitization process, i.e. red for the first image, green for the second image, etc.

9.2.5 DISPLAY

Load an image into the refresh memory for display. The image must be 512 x 512 or smaller to be displayed. A scrolling option is provided for pictures with more tnan 512 lines.

9.2.6 ENTER

The ENTER display function allows the user to enter an image directly from tape to the display. The ENTER function will automatically subsample images which are greater than 512 x 512 so that they can be displayed or magnify images that are smaller than 512 x 512 so that they will fill the entire screen.

All images are assumed to be in sample, bands, lines organization for display from tape. A keyword parameter is provided to allow the input of an image in samples, lines, bands organization.

Once the tape has been mounted the system keeps track of which file the tape is on so that subsequent requests to use the tape can be handled without requiring an operator response. If the user is done with the tape the CLOSETAPE function is used to unload the tape and free the tape drive.

All images used by the ENTER function must be in byte data type (8 bits per pixel).

9.2.7 FAST'DISPLAY

Fast display of a 512 x 512 pixel image from disc to display. The image must have an SLB organization, be of 512 x 512 size, and have BYTE data type, no spatial subsectioning is allowed.

9.2.8 FEEDBACK

This is the basic function that drives the Model 70 Feedback Subunit. Its inputs are the various pipeline outputs rather than named images. These pipeline data streams, after being transformed by the OFM's, are directed through the FEEDBACK/ALU into newly allocated refresh memory channels. The user can select any one, or all of the OFM outputs for FEEDBACK. The user is responsible for generating any LUT or OFM loadings in the pipeline. This function merely stores the results of an unspecified operation back into refresh. The function will, however, allow the user to use the IFM to rescale the pipeline outputs before they are returned to refresh. In addition, the function allows the use of a BLOTCH region to enable/disable the pipeline feedback, permitting the user to feed back only within, or only outside, a

given region of interest. The user is warned to use caution, particularly in smaller Model 70's, that the operation does not overwrite an important image channel. When a color FEEDBACK is being executed, it is possible to be modifying the pipeline inputs with feedback outputs such that succeeding feedbacks will be incorrect. The user is advised to verify that free or destroyable channels are available before executing this function.

9.2.9 LANDSAT

This function is used to input an image from a LANDSAT tape directly to the MODEL 70 Display. The LANDSAT tape can be in either the Sioux Falls or Goddard format. The default format is assumed to be 4 image files and the SIAT file on one 1600 bpi tape.

The LANDSAT function can both subsample and subsection the LANDSAT image as it is being display. If the Scroll keyword parameter is selected the image can be scrolled as it is displayed.

9.2.10 MAP

This function is used to read and display on the Model 70 an image file of coastal boundary data created by the CPU ">MAP" function from the CIA World Database file.

9.2.11 SAVE

SAVE is used to save an image stored in refresh memory (or memories) as a permanent disc image. The data is read out and saved in samples, lines, bands organization. The range of the data is determined by the number of bit planes configured with each refresh memory in the system. That is, a 6-bit system will generate data ranging from 0-63, an 8-bit system, 0-255. The output images are compatible with the fast display program FDISP.

9.3 DISPLAY ARITHMETIC MODULES

The following paragraphs describe the modules used to perform image arithmetic.

9.3.1 ADD

Performs pixel by pixel addition of any number of display channels. ADD computes the sum of N input images on a pixel by pixel basis. The user may specify weighting factors for each input image. This operation is performed in the Model 70 pipeline processor and does not alter the contents of refresh memory.

9.3.2 DIVIDE

DIVIDE computes the quotient of two pictures on a pixel-by-pixel basis. This operation is performed in the Model 70 pipeline processor and the original data is not destroyed.

9.3.3 MULTIPLY

MULTIPLY computes the product of N input images on a pixel-by-pixel basis. This function is performed in the Model 70 pipeline processor and does not alter the input data.

9.3.4 SUM

This function interactively sums two images of equal bands under trackball control. The horizontal position determines the relative weighting of the two images being added. When the cursor is all the way to the left the first input will constitute 100 percent of the sum. The weighting changes linearly until at the right side the second input will constitute 100 percent.

The vertical position controls the overall brightness.

The buttons have the following meanings:

A) Negate first input.
B) Negate second input.
C) Lock/unlock intensity variation (Y-direction)
D) Exit

9.4 DISPLAY IMAGE EXAMINATION MODULES

Modules used to examine displayed image data are described in the following paragraphs.

9.4.1 BILINEAR'ZOOM

This command performs an interpolative zoom of a displayed image. The magnification may be two, four, or eight. The input image may contain any number of bands.

9.4.2 COLORS

COLORS allows the user to define a color table interactively. Use function STASH to retain the color table on disk. Use function FETCH to restore a previously stored color table.

9.4.3 FLICKER

FLICKER allows the user to flicker between input images. Two flickering modes are provided.

The carousel mode allows the user to step through the images one at a time each time he presses a trackball function button. Depressing the bottom A allows the carousel to move forward. Depressing the button B allows the carousel to move backward. Press button C to enter the free-running mode and button D to quit.

In the free-run mode, FLICKER steps sequentially through the input images at a rate specified by the samples-position of the cursor. The leftmost position corresponds to the slowest rate, the rightmost position corresponds to the fastest rate. The trackball, used to position the crosshairs, may be rolled left or right while in this mode. Press the button, C, again to return to carousel mode. Function, FLICKER starts out in the carousel mode.

9.4.4 LEVEL'SLICE

This function interactively colors a range of pixel values on a display image. For an image with eight bit pixel values, an individual pixel can assume any value from 0 to 255; any subset of this range can be selectively colored with this function. Through the use of the trackball cursor position, the user can interactively modify the width of the color slice with the vertical position of the cursor and the start of the color slice with the horizontal location of the cursor. The width and location of the color slice is displayed in the upper left corner of the displayed image using graphics overlays zero and one. This function is used with one band images only.

Minimum hardware requirements: two graphics planes.

9.4.5 PALETTE

The Palette function allows the user to generate an infinite variety of pseudocolor Lookup Table loadings and then shift the loadings around in the table to highlight image features. It defines its own cursor by stores and returns the existing cursor on exit. It also allows the user to operate on an existing LUT loading rather than starting with a ramp and creating a specific loading. To fully understand its capabilities, the user is advised to practice with it.

9.4.6 PSEUDOCOLOR

This function generates a color map for display image on a pixel intensity basis. The algorithm uses hue, saturation and lighteness space to define colors. In this space equal steps of hue are supposed to provide equally perceptable colors. The validity of this hypothesis varies from observer to observer due to the extreme variations in the human observers perception of color. The colors are defined by a helix inscribed within a cylinder (height equals lighteness, the radius equals saturation,

and degrees of arc corresponds to hue) such that a particular color is defined for each pixel intensity.

If requested, the user can adjust some of the function parameters by moving the trackball. The buttons have the following meaning:

A) Lock/Unlock hue on X-axis
B) Lock/Unlock Y-axis
C) Toggle Y-axis control on steps/loops
D) Exit

The x-axis controls the hue parameter and the y-axis controls either the NSTEPS or NLOOPS parameter depending on a switch controlled by button "C." Using buttons "A" or "B" allows the user to lock the cursors movement in one direction and thereby freeze that parameter.

9.4.7 ROAM

ROAM in an interactive display function that enables the user to access and display 512 x 512 subsections from a mosaiced image that has been preloaded into refresh memory and that is larger than 512 x 512 pixels. The mosaiced image will usually be loaded into NxM >1 refresh memories.

9.4.8 SPLIT'SCREEN

This function enables the user to split the screen between either two or four images, depending on the input specifications. If two images are specified on input, then the user can interactively view a half screen window of the same portion of each image, the window being determined by cursor position. For the split between two images, the trackball button options provided zoom, horizontal or vertical split, and image swapping. If either four images or one, four band image are specified on input, the screen will be split into quadrants with either one image in each quadrant or one band of the four band image in each quadrant. As in the two-image split, the images may be scrolled by moving the trackball cursor, thus enabling the viewer to see the same portion of each image simultaneously. For four images, the trackball buttons allow the user to zoom the images, rotate the four quadrants, or swap the top two images. With these last two options, the user can interactively select the quadrant in which any or all of the images are displayed.

9.4.9 TMAG

TMAG magnifies, by replication, a displayed image (if NIDS=1) or a disc image (if NIDS=2) and updates the refresh memory with the magnified section. A density slicing option is allowed, with the results being overlayed over the image in graphics plane 0.

9.4.10 VIDEO'LOOP

This program allows the user to flicker between an array of images each assembled in one or more refresh memories. This is a useful technique for comparing images, particularly when the images represent a time-sequence series of some phenomenon such as weather patterns.

Four images are stored in each refresh memory for use with this function using the MOSAIC parameter in the display command. Each image therefore has a resolution of 256 x 256 pixels.

The function can operate in two modes. First, the function can loop through the images sequentially until the last image is displayed and then start again at the first image and repeat the sequence. Second, the function can loop through the images sequentially until the last image is displayed and then switch modes and cycle through in reverse until we reach the first image. The individual images can also be annotated with the frame number if the flicker rate is slow enough.

The flicker rate between scenes is controlled by the X-position of the trackball-controlled cursor. If the cursor is set at the left edge of the monitor, the flicker rate will be slow. If the cursor is set at the right edge of the monitor, the flicker rate will be fast. Actual flicker rates vary depending upon the loading of the host CPU.

If two or more input images or two or more refresh memories are specified, then the subimages in the second channel are displayed after the subimages in the first, and so forth.

9.4.11 ZOOM

This program allows an interactive magnification process of the data by up to 8X magnification via the cursor. This program can run only on the M70E which has the hardware zoom.

The menu commands are:

Button A = Toggle Button Mode
Button B = Toggle Trackball Mode
Button C = Increment Zoom Factor
Button D = Quit

The user specifies the center point of the data to be magnified by moving the cursor to that position, sets the magnification factor, then iterates until thru.

9.5 DISPLAY SUBIMAGE DEFINITION MODULES

Modules that perform operations on subimages are described in the following paragraphs.

9.5.1 BLOTCH

BLOTCH is used to define irregular image subareas in use with the videometer or ALU. The analyst may specify an unlimited number of regions. BLOTCH thereby enables him to build up arbitrarily shaped polygonal regions for his particular application. The various uses of this function are shown below.

9.5.2 CNPT

This program accepts as input 2 display files, and enables the user to generate a file of control points that he designates using the trackball. Pushing button "B" cycles the display between the two input display files. Pushing button "A" designates a control point. Pushing button C toggles between a zoomed presentation and the normal presentation. The use of the zoom mode allows user to select individual pixels easily. Exactly one output file should be specified. The "control point file" can be used as an input to the programs WARP and FRAME. The sequence of the two display files input to CNPT should be so that the first is the file that the user wishes to warp to conform to the second.

A previously generated control point file may be used as a third input to CNPT in order to accumulate control points from several control point picking sessions. This feature allows the user to effectively pick control points for images larger than 512 x 512 pixels.

By pressing the D button, the user may enter an interactive edit mode in which he can delete points or fit a transform to the points selected.

9.5.3 DECODE

This function plots a variety of polygons in refresh memory from a vertices file previously made by "^ENCODE" or ">POLYGON'ENTER." In the same way that ^PLOT is the opposite of ^BLOTCH, this function is the opposite of ^ENCODE. It accepts display images and a vertices file outputted by ENCODE or POLYGON'ENTER, and plots each of the polygons as a distinct number in refresh. If the polygons have associated nesting levels, they will be plotted in order of nesting level, where 0 is the outermost level. Nesting levels do not apply in ENCODE, but do in POLYGON'ENTER. Polygon values range from 0 to 255. The polygons may be labeled with their values in graphics, if desired.

9.5.4 ENCODE

This function is much like the BLOTCH function except that it writes a specified value (between 0 and 255) into an 8-bit refresh memory instead of merely filling a 1-bit graphics plane. ENCODE allows the user to "paint over" regions in an image,

digitize thematic map data, or clean up spurious noise in an otherwise homogeneous area of an image. In particular, it was designed to help merge image data with thematic map data. When used in conjunction with DIGITIZE, the user can record a map as an image with the video digitizer and then ENCODE thematic data over this map image, much like "paint-by-numbers." An optional output vertices file may be specified if the user wishes to save these regions and apply them to another image (see ^DECODE).

9.5.5 PLOT

This function allows the user to review training samples generated by the display function (^)TRAIN. A display resident image is provided as input along with one or more vertices files that describe the locations of various training areas or "class paradigms." The user can then zoom in and look at the areas selected for each class, optionally examine the statistics of those areas using the display hardware, and alternately blank out either the rest of the image or flash the graphics plane to fully understand the training areas and their accuracy.

9.5.6 TRAIN

This function is used in the feature extraction process of supervised multispectral analysis. Blotch vertices and class names are accumulated for each training region defined by the user and stored in the output disc file. This file of vertices can then be used as input to other functions.

9.6 DISPLAY IMAGE STATISTICS MODULES

The modules used in performing statistics on display images are described in the following paragraphs.

9.6.1 HECTARE

HECTARE computes the number of hectares in an image for each class in a supervised or unsupervised classification image. A table containing the class number, number of hectares per class, the percent coverage, and the number of pixels per class is output. For supervised classification the class names are also printed in the table.

An option allows the analyst to limit the hectare analysis to a previously selected blotch region (a set of polygons defined over the displayed image). Normally, the entire displayed image is analyzed.

The input requirements are:

If there are two inputs, the first input must be a single band of the displayed image and the second input must be a statistics file output by the function PREPARE (E.G. $A PREPARE).

9.6.2 HISTOGRAM

HISTOGRAM uses the videometer to compute the histogram of an image in the Model 70 refresh memories. When no image is input the area to be histogrammed is selected by specifying the upper left and lower right corners of the videometer marker. REGARDLESS OF INPUTS, the user may select the BLOTCH option that uses a blotch image created by the BLOTCH function.

One histogram is generated for each refresh channel contained in the image. If no input image is provided to the HISTOG function then the image being displayed will be histogrammed and the resultant histogram is displayed on the graphics overlay. The function assumes that the image is a black and white image so if a color image is being displayed, the user must specify the COLOR parameter.

9.6.3 LINEAR'FEATURE

This function assists a geologist in annotating and summarizing linear features that are visible in a scene. Three basic capabilities are provided by the function. The user can manually delineate linears with the trackball; the user can summarize those linears in a rose diagram; and the user can direct a summary table of distances and orientations to the terminal or lineprinter. The distances used in the program can be in kilometers, miles or any other unit by changing the SCALEFACTOR parameter.

The summary rose diagram is divided in half. The left side shows each identified linear by its orientation and length. The right side summarizes the number of linears that fall into 15 degree groups by orientation. This is done by means of a "pie" diagram where each pie section's length is proportional to the number of linears whose orientation fell into that 15 degree range. The units of distance in the summary diagram are shown in the bottom center of the graph.

In Model 1, the Trackball Button Options are:

Button A) Pick a point
B) End linear
C) Increment zoom
D) Go To Mode 2

In Mode 2, the Trackball Button Options are:

Button A) Go to Mode 1
B) Display image & linears /
Summarize linears in graph /
Display image only
C) Erase selected linear(s)
D) Exit function

9.6.4 POINTS

Use POINTS to display the line, sample, and intensity value of the pixel at the current position of cursor. The readout changes whenever the "A" button on the trackball assembly is pushed, or when in free-running mode, whenever the trackball is moved.

9.6.5 PRINT

This function is similar to ^POINTS in that it prints on the screen brightness values from an image in refresh memory. It differs in that it prints the values for a 16 x 16 window of the image in an entire screen of graphics. When a three band, color image is input to the function, it will print the window of brightness values in their respective colors (red-green-blue). The sample and line numbers are printed around the edges. The following button menu is used:

A - Print a new window
B - Toggle image and graphics
C - Toggle zoom (1x - 4x - 1x ...)
D - Exit

If the function is working on a nonsubsampled 16 x 16 window, then it will redraw the cursor to give a 16 x 16 box shape. Then the user can see exactly what size window will be printed. In addition, if the user zooms using button "C" and the INCREMENT parameter is still "1 1" then the function will redraw the cursor to show a 64 x 64 box which would be equivalent to 16 x 16 zoomed 4x. If the cursor box would end up larger than 64x64 at any time, then the function will revert back to a crosshair cursor.

9.6.6 PROFILE

This function allows the user to interactively plot an intensity profile along a user specified line. The line segment is defined by the user via the trackball, and written on graphics plane zero. An intensity profile is plotted for each band of the input image on graphics plane one.

9.6.7 SCATTER

The SCATTER function plots a bivariate frequency distribution of the two display-resident channels provided as inputs. The plot can be placed in either a graphics plane or a full 8-bit refresh memory channel. The plot can be limited to those pixels contained in a region of interest in another graphics plane and it can be scaled by a given scaling factor. Marginal histograms can be plotted of each individual channel, either to the top or the side of the bivariate plot. The function uses the display hardware, notably the videometer, in an efficient manner causing the function to run in only a few seconds.

9.6.8 STATISTICS

This function produces a printout of summary statistics for any display resident image covering the entire image, a subregion defined by ^BLOTCH, or the complement of that subregion. The contents include the following:

- Number of pixels considered.
- Minimum value of the pixels encountered.
- Maximum value of the pixels encountered.
- Mean value of the pixels.
- Standard Deviation of the pixels.
- Mode of the pixels.
- Median value of the pixels.
- Values of the three Quartiles.
- Values of the nine Deciles.

This function can be used to explore various training areas for supervised classification, or it can used to design radiometric enhancements of an image. For example, the deciles could be used with the function ^PLIM as breakpoints to approximate transformation functions such as COSINE not yet available in the System.

9.7 DISPLAY RADIOMETRIC TRANSFORMATION MODULES

Modules used to perform radiometric transformations on images are described in the following paragraphs.

9.7.1 ADJUST

This function allows the user to radiometrically transform an image so that the results have a user-specified mean and standard deviation. The operation is equivalent to shifting and spreading (or compressing) the histogram of the image. The input statistics used in this function are obtained from either the entire image or a region of interest defined in a given blotch plane. When the desired change of the standard deviation involves an expansion of the distribution beyond the range of the Model 70 (i.e. 0 - 255) a warning will be printed because the results may not be as requested. Not that negative values from the lookup-tables, although physically possible, are not generated because of possible difficulties with later functions.

9.7.2 EXPONENTIAL

EXPONENTIAL applies an exponentially shaped intensity mapping to the input images using the lookup tables. If the input image has three channels, the image will appear in color; otherwise, the channels are summed and displayed in black and white.

9.7.3 HIST'EQUALIZE

HISTEQ computes an intensity mapping from the image histogram producing an image with a linear cumulative d.f. histogram. If the input image has three channels, the image will be displayed in color, otherwise the channels are summed and displayed in black and white.

9.7.4 HIST'MATCH

This function performs a radiometric operation to match the histogram of one file to another. The first input image is transformed to meet the second. Two refresh memory inputswith the same number of bands are needed.

9.7.5 HIST'NORMALIZE

This function performs a histogram normalization on the input image. The original image is not affected. A marker-mode histogram of the result will show a more normal distribution of the pixel intensities. The result may be catalogued by using the FEEDBACK function. The PIPELINE'PLOT function can be used to see how this function works.

9.7.6 LOCAL'ENHANCE

LOCAL'ENHANCE performs a space-variant contrast stretch on a display image (i.e. an image resident in Model 70 refresh memory). Since the processing takes place directly in the algorithm, abort it if desired, and evaluate the effect of various parameter combinations quickly. The processing is either done in place (written over the original display image), or written into another refresh memory (or memories). If desired, the resultant display image can be saved as a permanent disc image by use of the program "SAVE." Pushing either of the trackball buttons will abort the program. The algorithm works as follows:

$$X' = \frac{S'}{S + MAX/S} * (x - M) + A*M' + (1-A)*M \tag{1}$$

where x is the original pixel value at a given sample-line location, S is the "local" standard deviation, defined in the immediate NxN neighborhood (about X), M is the local mean value (similarly defined), M' is the desired mean value, and S' is the desired standard deviation. If MAX=0 and A=1, Eqn. 1 reduces to

$$x' = (S/S') * (X - M) + M' \tag{2}$$

which merely shifts the local statistics to the desired mean and standard deviation. The purpose of the "MAX" and "A" parameters are to restrain the rather harsh filtering effect of Eqn. 2. MAX is the maximum gain, and "A" is a parameter which governs the

mean value shifting aspect of the filter. (If A=1, the mean becomes M', if A=0, the mean remains M.)

9.7.7 LOGARITHM

LOG applies a logarithmically shaped intensity mapping to the input image using the lookup tables. If the input image has three channels the image will appear in color; otherwise, the channels are summed and displayed in black and white.

9.7.8 NEGATIVE

NEGATIVE applies a linear intensity mapping that inverts the intensity range. If the input image has three channels, the processed image will be displayed in color; otherwise, the channels will be summed and displayed in black and white.

9.7.9 PIECEWISE'LIN

PIECEWISE'LINEAR performs a piecewise linear intensity mapping using a list of user defined break points. Or, if no list is provided, the user performs an interactive PIECEWISE'LINEAR via cursor-specified breakpoint coordinate pairs. If the interactive method is chosen, a normalized cumulative histogram is obtained from all input image bands specified by the user, and plotted on a graphics plane on the top half of the display screen. On the bottom half a coordinate system is then displayed with the vertical axis being the "TO" quantization values and the horizontal axis being the "FROM" values. BUTTON=A chooses a new pair, BUTTON=B deletes an old pair, BUTTON=C quits. The user moves the cursor to a new breakpoint pair on the coordinate system, and chooses which button to push. Consecutive pairs are connected with straight lines showing the mappings of all points between points. When the user quits, the PIECEWISE'LINEAR mapping is displayed. If the input image has three channels, the processed image is displayed in color; otherwise, the channels are summed and displayed in black and white.

9.7.10 SCALE

SCALE computes a linear intensity mapping from the histogram that maps the minimum value to black and the maximum value to white. The user may alternately specify clip levels. If the input image has three channels the scaling will be done to each color separately; otherwise, the images will be summed and scaled in black and white.

9.7.11 TLM

This function allows the user to interactively manipulate a positive or negative linear intensity transformation of the input image by changing the cursor position.

The trackball button options are:

Button A) Positive picture
B) Negative picture
C) Toggle function display
D) Quit

9.8 DISPLAY MULTI-BAND SPECTRAL TRANSFORMATION MODULES

Modules used for spectral transformation of multi-band images are described in the following paragraphs.

9.8.1 FFT1D

"FFT1D" performs a discrete fourier transform on the spectral dimension of a four channel display image and displays the results in the RED, GREEN and BLUE image channels. The GREEN channel contains the DC component, the RED channel contains the real part of the first frequency and the BLUE channel contains the imaginary part of the first frequency. The highest frequency is not displayed.

The user may select clip levels which will be used to scale the results of the transformed image.

9.8.2 HADAMARD

"HADAMARD" performs a Hadamard transform on the spectral dimension of a four channel display image and displays the results in the RED, GREEN and BLUE image channels. The GREEN channel contains the DC component, the RED channel contains the first sequence and the BLUE channel contains the third sequence. The highest sequence is not displayed.

The user may select clip levels which will be used to scale the results of the transformed image.

9.8.3 KL'TRANSFORM

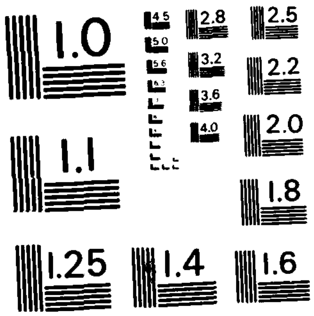
This function performs a Karhunen-Loeve transform on an image in the Model 70 and displays up to three components on a color monitor. The input image must have at least three bands. The image statistics file must be supplied and can be made using >STATISTICS. The form of the Karhunen-Loeve transform used is:

$$\mathrm{OUT(comp)} = \sum_{\mathrm{band} = 1}^{\mathrm{NBANDS}} \mathrm{EIGEN(band,\ comp)} * [\ \mathrm{IN(band)} - \mathrm{MEAN(band)}\ ]$$

AD-A130 779 A DESCRIPTION OF THE NORDA SATELLITE REMOTE SENSING HARDWARE/SOFTWARE FOR..(U) NAVAL OCEAN RESEARCH AND DEVELOPMENT ACTIVITY NSTL STATION MS.. G D STEPHENSON 2/2

UNCLASSIFIED APR 83 NORDA-TN-205 F/G 9/2 NL

END
DATE
FILMED
9 83
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

This transform is equivalent to a principal components, feature space rotation of an image. Each component will contain a percentage of the overall variance equivalent to its eigenvalue in the statistics file. The output is three selected components displayed as red, green and blue on the display. No permanent output is produced but the display can be fed back (in color) to save the results of this function. The function cannot produce more than three components but it can accept up to twelve channels of input and display any three of the output components.

9.8.4 MATRIX'TRANSFORM

MATRIX'TRANSFORM performs a linear transformation from N image bands to the 3 drive signals (red, green, and blue). By appropriate loadings of the lookup tables the following generic transformation is applied to the refresh memory data:

$$\mathrm{RGB}(I) = \sum_{J=1}^{N} \left(\mathrm{MATRIX}(I,J) * \{\mathrm{BAND}(J) .+ \mathrm{BIAS}(J)\}\right)$$

Where:

N = the number of bands in the image.
RGB(I) = red, green, blue for I=1,2,3.
BAND(J) = the image in the Jth band.
BIAS(J) = an arbitrary additive bias.
MATRIX(I,J) = an arbitrary 3XN matrix of weights.

The red, green, and blue data signals are automatically scaled by using the videometer to extract the histograms of the data from the raw transform. Consequently, the red, green, and blue components will appear to be "turned on" sequentially.

9.8.5 RATIO

RATIO computes the ratio of adjacent spectral bands for a four channel input image. The output may be scaled by user specified clip levels. RATIO computes image band(1)/band(2), image band (2),/band (3), image band (3)/band (4), and outputs them to the red, green, and blue display channels respectively.

9.8.6 SLANT

"SLANT" performs a slant transform on the spectral dimension of a four channel display image and displays the results in the RED, GREEN and BLUE image channels. The GREEN channel contains the DC component, the RED channel contains the first sequency and the BLUE channel contains the third sequence. The highest sequence is not displayed.

The user may select clip levels which will be used to scale the results of the transformed image.

9.9 DISPLAY SPATIAL TRANSFORMATION MODULES

Modules used to perform spatial transformations on images are described in the following paragraphs.

9.9.1 CONVOLVE

This function performs a two-dimensional spatial convolution on a one band display image utilizing up to a 256 element kernel specified by the user. The algorithm utilized is of the following form:

$$p'(x,y) = \sum_{i=-m/2}^{m/2} \sum_{j=-n/2}^{n/2} p(x+i,y+j) * weights(i,j)$$

where P'(x,y) is resultant pixel intensity after convolution x is the line position of the pixel, y is the line number of the pixel, P(x,y) is the original pixel intensity, WEIGHTS is an n by m array containing the convolution kernel, n is the number of kernel elements per row, and m is the number of kernel elements per column.

9.9.2 EDIT'IMAGE

This routine interactively filters dropped lines or pixels noise in a displayed image via TUKEY filtering. It is done on a pixel, line, or column basis. Results are displayed in color for 3-band images, and b&w for non-3 band images.

This program works only on M70E displays which have the hardware zoom feature. User is prompted with cursor menu description on the terminal.

9.9.3 FFT2D

This command computes the two-dimensional Discrete Fourier transform. Note that the number of lines and number of samples must both be an integral power of 2 for this function. The FFT2D function will not handle an image larger than 512 x 512. This function requires four output channels and therefore requires a minimum of five channels of refresh memory in the Model 70. This function also requires the 32 bit MUX feature of the Model 70 interface adaptor which may not exist on older units.

9.9.4 GAUSS'FILTER

Generates and applies a Gaussian shaped frequency domain filter to a complex type image previously Fourier transformed by ^FFT2D.

9.9.5 IFT2D

Performs the inverse two-dimensional spatial Fourier transform of a data set of the form produced by FFT2D. This function requires a minimum of 5 channels of refresh memory, four for the input image and 1 for the output.

9.9.6 LIFE

This function implements the game of life invented by John Horton Conway as described in Martin Gardner's column in Scientific American (October 1970 and February 1971). Successive generations of an initial kernel are computed using the feedback/ALU option of the Model 70.

9.9.7 MEDIAN'FILTER

This command performs a two-dimensional TUKEY median filter of a displayed image. The user specifies the number of rows and columns in the filter kernel. The kernel may be explicitly specified or one of several predefined kernels may be employed. The output may be saved in place or written to new channels.

9.9.8 REGISTER

This function uses the scrolling and feedback capabilities of the Model 70 to register one input image to another producing a registered copy of the first input. The user picks a point on the first input, links the trackball to it, and then interactively moves this linked image over the second input until it is satisfactorily registered. On exit, the shifted input image is fed back into output channels(s).

9.10 DISPLAY GEOMETIC TRANSFORMATION MODULES

The following modules are used to perform geometric transformation on the display image.

9.10.1 MAGNIFY

This function performs a magnification (or minification) of a refresh image by user specified samples and line factors. The output image may be truncated to fit into refresh memory. This function uses Model 70/F feedback and scroll.

9.10.2 ROTATE

This function rotates the input image by a user specified angle using the Model 70/F scroll and feedback. It is faster than the CPU ROTATE function but the input and output images must be in refresh memory.

9.11 DISPLAY CLASSIFICATION MODULES

The following modules are used for classification functions on display images.

9.11.1 CLUSTER

This function performs an unsupervised classification on a display-resident image at extremely high speed because it uses a variety of hardware features in the Model 70. By unsupervised we mean a classification that does not involve the use of training samples or procedures. The classifier makes a statistical estimate of seed locations in the feature space for a user specified number of classes, performs a classification, examines the results, modifies the location of the seeds, and classifies again. The process of classification--evaluation--modification is repeated for a user specified number of iterations. The process should "home in" on a stable classification that tries to divide the feature space into a clear set of classes. The initial number of classes is specified by the user but other parameters can be specified that will cause classes to be merged or be split. The user can specify a "quit" parameter which will cause the function to stop when the improvement of each successive classification is less than a certain threshold. The user can also view the interim results of each iteration and manually quit if the results are adequate. An option of selecting between brief and detailed tabular results is available and these results can be listed on the lineprinter as well as the terminal. The user can force the program to generate its own initial seeds or can provide statistics from a disk file as output by >PREPARE, ^MINDIST or a previous ^CLUSTER operation. The function will optionally save the statistics used by the last iteration in an output disc file for use by subsequent ^MINDIST and ^CLUSTER runs.

9.11.2 MINDIST'CLASSIFY

MINDIST is a hardware implementation of a minimum distance classifier. The image to be classified must be display-resident and can theoretically have up to ten bands. In fact, the number of channels available in the user's Model 70 determines the maximum number of bands that can be used in the classification. The function requires two bands to hold its interim and output results, so in the case of a six channel Model 70, the classifier is limited to four channel inputs. The computations involved in the classification process are performed in the pipelines of the Model 70 at a rate of approximately two classes per second (for a

typical 4-band image), and so a 24-class operation should take around 12 seconds. The function can accept input statistics for each class in three different ways, and can be instructed to save these statistics in a file on exit. Statistics can be input as follows:

- By providing a statistics file like that generated by by the >PREPARE, ^CLUSTER, or a previous ^MINDIST function.

- By going through a simplified training process in this function which allows the user to define blotch areas around training samples, and then extracts the necessary statistics for each band.

- By manually typing in a set of mean and standard deviation vectors for each class.

The function accepts all the inputs as images but can be instructed by a parameter to interpret the last input as a statistics file. If the command line includes an output file then the statistics used in the classification process will be written to that output file in an abbreviated form that can then be used by some of the other classifiers in the system. This function does not, however, produce a complete statistics file because it does not need to generate all of the statistics needed by a maximum likelihood classifier. This function should not be considered, therefore, as a primary training input into the system. The ability to write statistics to files is provided as a convenience to this and other display functions and is not comparable to the normal training output.

9.11.3 TABLE'LOOKUP'CLASSIFY

This function performs a table driven supervised classification. The user may interactively select up to three training areas via the trackball on a 2, 3, or 4 band display image with the resulting classes displayed in red, green, and blue respectively. The classification is performed by setting up the Model 70 lookup tables in such a way as to quantize each of the spectral channels, thereby mapping each pixel to a unique location in an n-dimensional space. This point in n-dimensional space is then input to the output function memory (OFM's) where the decision function is implemented. This decision function is derived from the quantized n-dimensional histogram of the training area. The most populated training cells will be mapped to full brightness, the least populated cells to half brightness, and the unoccupied cells to black. After each class is chosen, and the resultant classification performed, the user proceeds to the next class by pressing any trackball button. If the user specifies the print option, the percentage of coverage and the number of hectares in each class will be output to the line printer.

9.11.4 VEGETATION

The VEGETATION function is used to obtain a quick estimate of the number of hectares of vegetation contained in an image. The hectarage estimate is obtained by subtracting the green band from the IR Band in a conventional Landsat image and displaying all resultant levels that exceed a user specified clip level which is defaulted to 1 as red which indicates vegetated areas.

9.12 DISPLAY GRAPHICS MODULES

The following modules are used to perform operations on the display graphics.

9.12.1 ANNOTATE

ANNOTATE allows the user to interactively annotate an image in refresh memory or in the graphics overlay. If no inputs are given, the image currently displayed will be annotated and annotations will go to graphics no matter what. If there are inputs, annotations will go to refresh memory by default, unless the parameter "GRAPHICS" is specified.

There are 2 modes of operation. In mode 1 (definition mode), the trackball menu applies, and the user is allowed to change various parameter definitions. In mode 2 (positioning text mode), the user selects annotation position. If annotating to graphics, the user moves the cursor to the position he wishes the text to be, and then presses any button. If annotating to refresh, the text will be the cursor, so the user just moves the text to the desired position.

ANNOTATE's trackball menu is:

The trackball button options are:
Button A) Do annotation
B) Change characteristics
C) Change color scheme
D) Exit

When first entering the function, after the parameter prompts, the user will be in mode 2 (positioning the text). The message:

Select cursor position, then press any button.

will cue the user that he is in mode 2. When done selecting text position, the message:

Select option from menu

will cue the user that he is in mode 1 (definition mode). Button A will prompt for the annotation text and then go to mode 2.

Button B will prompt for parameters ASPECT, BOX/CHARVALUES, CURSORPOSITION, FONT, HEIGHT/VSPC/HSPC, and SKEW, and leave the user in mode 1. Button C will prompt for BACKGROUND and COLOR (these parameters only affect graphics), and leave the user in mode 1. Button D will exit the user from the function.

9.12.2 BLANK'GRAPHICS

This program blanks user-specified graphics planes. Note: It not only disables the display of graphics but also erases the contents of the graphics refresh memory.

9.12.3 BOUNDARY

This routine determines the boundaries of homogeneous regions defined by a graphics plane, and zeroes the interior regions leaving only the borders of the homogeneous regions.

9.12.4 CONTOUR

This function allows the user to generate a contour "map" of the input display-resident image. It loads the lookup tables to display a series of evenly spaced isolines over the image. The coherence of the results of this function depend on the "smoothness" of the image. A highly variable image will produce noisy contour lines while a continuously varying image's contours will appear more realistic. In some cases the user would obtain better results by first smoothing the image with the convolution function. Inclined contours can be produced by adding to the input image a synthetic greyscale of the desired inclination before contouring. After the contours have been generated the user is provided with the opportunity to interactively label the contours with the trackball.

9.12.5 CURSOR

This command sets the programmable cursor to the user specified color, shape, and blink rate. This function reads the contents of the cursor, modifies it to include the selected cursor size and shape, then rewrites the cursor. This allows the user to build a variety of cursors. The BLANK parameter is used to erase any previous cursor.

9.12.6 DRAW

Using this function, the user can graphically annotate images by drawing on them. The function is particularly designed for use with the tablet hardware option to the Model 70. This option allows the user to literally draw on the screen by pressing the pen to the tablet. With the normal trackball the function is still capable of annotation but it is clumsier. The user can interactively change colors of his "ink" and erase mistakes. The

function can also be used over existing graphic information from other functions like ^ANNOTATE.

9.12.7 GFLICKER

GFLICKER allows the user to flicker between user specified graphics planes. Two flickering modes are provided.

The carousel mode allows the user to step through the graphics planes one at a time each time he presses a trackball function button. Depressing the left button allows the carousel to move forward. (If REPLACE, the graphics planes are superimposed and all previously displayed planes on a given cycle remain on when proceeding in the forward direction.) Depressing the right button allows the carousel to move backward and causes the last plane displayed to be turned off, or if no planes are displayed, the last plane in the sequence to be turned on.

In the free-run mode, flicker steps sequentially through the graphics planes at a rate specified by the samples-position of the crosshairs. The leftmost position corresponds to the slowest rate, the rightmost position corresponds to the fastest rate. The trackball, used to position the crosshairs, may be rolled left or right while in this mode.

Function GFLICKER starts out in the free-running mode. The default initial mode is the free-running mode. Depressing both function buttons simultaneously allows one to exit from any mode. If one is in the carousel mode and both buttons are pressed, one will enter the free-running mode. If one is in the free-running mode, either the program will stop or (if REPROMPT=TRUE), the analyst will be reprompted for the program parameters.

9.12.8 GRAPHIC'FEEDBACK

This function uses the Model 70 Feedback Subunit to feedback a density slice of an image in refresh into a specified graphics plane. Its inputs are the various pipeline outputs rather than named images. These pipeline data streams, after being transformed by the OFM's are directed through the FEEDBACK/ALU into the INPUT FUNCTION MEMORY, which sends the selected density slice of the image to the specified graphics plane. The user can select any one, but only one, of the OFM outputs for GRAPHIC'FEEDBACK. The user is responsible for generating any LUT or OFM loadings in the pipeline.

9.12.9 GRAPHICS'ON

GRAPHICS turns on any or all of the graphics planes and selects the appropriate color for each graphics plane.

9.12.10 GRAPHICS'OFF

GRAPHICS'OFF turns off the graphics overlays, but does not blank them out. Contents from previous manipulations remain.

9.12.11 GRID

GRID overlays a grid on the screen using the graphics overlay. The spacing of the grid lines is specified by the user.

9.12.12 GSAVE

Used to save a copy of user-specified graphics planes on disc. Each plane is stored as a single band byte image.

Note that plane 7 containing the system status may be stored for future retrieval. Note also that blotches may be stored as images by using this function.

9.12.13 PIPELINE'PLOT

This function plots the contents of LUT's, OFM's and registers in 1 or 3 graphics planes to show the Model 70 status at the time of use.

9.12.14 THEMES

"THEMES" is used to display the results of supervised or unsupervised classification on the graphics overlay. The user may select up to seven themes to be displayed at one time. For each theme the user specifies the number of classes in that theme and the class numbers. Classes will be written in planes 0 thru the maximum number of graphics channels and will be colored BLUE, GREEN, RED, ORANGE, YELLOW, BROWN and VIOLET. The first theme will be BLUE the second GREEN and so forth. The user may use the GRAPHICS function to change the colors after they have been written in the graphics memory.

9.12.15 TICMARK

TICMARK uses the graphics overlay to place ticmarks around the borders of the image and, optionally, reseau marks every 50th column and line within the image. Ticmarks occur every 10 units with a major ticmark every 50 units.

9.12.16 WRITE'GRAPHICS

Used to display an 8-bit valued or binary image on the graphics overlay. This is of utility in superimposing maps or grids onto conventional images. If the input image type is multi-valued (i.e. not binary), the user is prompted for clip levels which are used to threshold the image and map all the pixels having intensities between the clip levels into 1 and all others

into 0 (or vice versa). This is useful for generating BLOTCHES which delineate certain regions in an image (use PLANE=6 and function SAVE'BLOTCH) and for density slicing operations.

9.13 DISPLAY CHARTING UTILITY MODULES

The following modules are used for charting functions on the image display.

9.13.1 BAR'CHART

This function produces additive or nonadditive bar charts of user-supplied values.

9.13.2 CHART

This function produces additive or nonadditive graphs of user-supplied values.

9.13.3 PIEPLOT

This function produces a pieplot from user-defined values. Each wedge in the circle will be labelled outside of the circle at the center of its arc by either the actual value entered for that wedge or by its value as a percent of the sum. A key for the plot is also generated.

9.14 DISPLAY THREE-DIMENSIONAL GRAPHIC MODULES

These modules provide three-dimensional graphic capability.

9.14.1 GPERSPECTIVE

This function plots a perspective view of a surface as a net of lines in graphics. The surface is input in the form of an image file where the x and y coordinates are represented by sample and line locations and the z coordinate is set to the intensity value contained in that pixel. The function removes hidden lines, where a closer section of the surface obscures the area behind it. It allows the user to exaggerate vertical changes in the surface, and provides a number of ways of specifying the viewing orientation and controlling the appearance of the output image. It will plot onto an existing plane without blanking it and it will permit panelling of large plots.

9.14.2 PERSPECTIVE

This function plots a perspective view of a terrain and colors it. It requires that the user provide two inputs: a one-band "surface" input and a multiband ($n<4$) "color" input. The surface input is an image file where the x and y coordinates are represented by sample and line locations and the z coordinate is set to the intensity value contained in that pixel. The color of

the surface at any pixel is defined by the values in the second input file. It is assumed that the surface input and color input are registered so that each pixel has an elevation in the first input and a color in the second input. The function removes the hidden surfaces, where a closer section of the terrain obscures the area behind it. It allows the user to exaggerate vertical changes in the surface, and provides a number of ways of specifying the viewing orientation and controlling the appearance of the output image. It will plot on an existing image if it is provided as a third input, and it will permit panelling of large plots.

9.15 DISPLAY TABLE DIGITIZE MODULES

These modules are used for tables digitizing functions.

9.15.1 TD'INITIALIZE

This function is necessary to integrate a Summagraphics ID-2 series table digitizer into the system. It establishes a mathematical transform to connect a map on the table digitizer to an image on the display. This is accomplished by selecting coincident control points on each device and thereby "training" the system to map table digitizer coordinates into the image coordinate system. After an initial training stage, the function will begin to predict the display coordinates of selected points. The training process continues until the predictions are sufficiently accurate to satisfy the user.

9.15.2 TD'MAP

This function allows the transfer of map data from a table digitizer to a graphics overlay. It requires a transform file as input, which can only be produced by the function ^TD'INIT. The user is given the ability to replace the trackball with the table digitizer stylus, with the function reading coordinates from the table digitizer, using the transform file to convert them to image coordinates, and moving the cursor to these points. The functions ^BLOTCH and ^DRAW are emulated by this function, but with cursor control coming from a registered map.

9.16 DISPLAY UTILITY MODULES

These modules are general utility modules for the display.

9.16.1 BUFFER

This function allows the user to disable/enable IO buffering to the MODEL 70. The user may also request a dumping of buffer values to the terminal.

9.16.2 CONSTANT

This is a quick function to produce a constant image of a user-selected color for use in ^FEEDBACK or simply to blank the screen (to make graphics more visible, for example).

9.16.3 CZCS'ENHANCE

The ENHANCE function applies the image enhancement functions contained in the CRTT image file output from CZCS.

9.16.4 CZCS'TEMPERATURE

The CZCS'TEMPERATURE function is used in conjunction with the CRTT file output from the CZCS function to display the temperature of a pixel selected interactively using the trackball and cursor. The CZCS'TEMPERATURE function is operated in the same manner as the POINTS function.

9.16.5 FETCH

This function will restore part or all the Model 70 tables and control registers associated with a given input from a disc file so that the system environment will be identical to a previously saved situation. If no parameters are specified, the entire thing will restored, else only selected portions as specified using the parameters will be restored.

9.16.6 KEY'CLASS

This function allows the user to add a key to a class map or any other single band display image.

9.16.7 PAUSE

This function pauses the command interpeter until a trackball button is pressed. A series of commands may be entered at once at the beginning of a session with PAUSE commands embedded allowing the user to run a pre-planned experiment with minimum intervention and still allow time for transcribing notes or statistics.

9.16.8 PROCESS

This function processes images larger than 512 x 512 using the Model 70 pipeline. The function performed is determined by a previously STASHed pipeline configuration. The number of bands in the process input image should agree with the STASHed pipeline configuration. If the STASHed configuration was for a one-band image, the process input image must be one-band. If the STASHed configuration was for a three-band image, the process input image must also have three bands.

9.16.9 STASH

This function will store all the Model 70 tables and control registers associated with a given input in a disc file so that the system environment can be restored later. Items stored include scroll registers, color assignments, range registers, sum processor constant registers, function look-up tables and output functions memories.

10.0 SYSTEM/UTILITY

The following programs were developed/modified for IDSIPS systems support.

10.1 PROGRAM/TECHNIQUES

10.1.1 PVCLEAN

Program to purge deleted files from user's private disc volumes.

10.1.2 TPHR

Program to read CCT tape header records and provide a formatted print of pertinent header record contents for user information.

10.1.3 TAPECAT

Interactive program to maintain a database of all CCT image tapes in the laboratory.

10.1.4 TDUMP

Program to provide octal dump in full word or byte format of digital tapes. Specific file, number of records to dump and conversion from DEC format to HP3000 format capability are additional options.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER NORDA Technical Note 205	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A Description of the NORDA Satellite Remote Sensing Hardware/Software for Prospective Users		5. TYPE OF REPORT & PERIOD COVERED Final 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Gerald D. Stephenson		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Ocean Research and Development Activity Ocean Science and Technology Laboratory NSTL Station, Mississippi 39529		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 6.3(704N)
11. CONTROLLING OFFICE NAME AND ADDRESS Same		12. REPORT DATE April 1983 13. NUMBER OF PAGES
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

IDSIPS	System	Overlays	DMSP	Display
Satellite	Computer	CCT	Calibration	CPU
I^2S	Model	TIROS	IPOS	Image Processing
Image	Graphics	GOES	Command	Satellite Receiving

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This document is intended to provide a general overview of the hardware and software used in the NORDA Code 335 Remote Sensing Branch. Detailed documention of the International Imaging Systems software can be found in their System 101 Image Processing System Users Manual.

A license agreement exists between NORDA and I^2S that prohibits NORDA from making this software available to other facilities. Any facility that is interested in using I^2S software would have to make a license agreement with I^2S.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

ATE

LMED

8